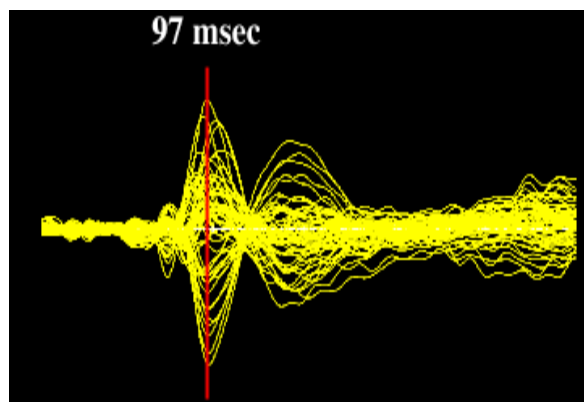
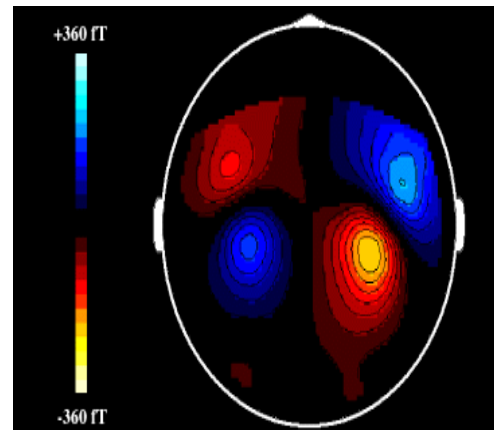
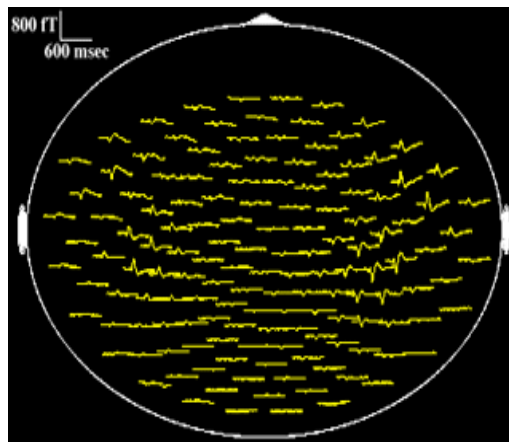


MEG CORE FACILITY

MEG Data Analysis Tutorial



MEG CORE FACILITY

MEG Data Analysis Tutorial

National Institutes of Mental Health (NIH)
9000 Rockville Pike
Bethesda, Maryland 20892
Phone: 301/402-2362

*This manual contains information and images from CTF Systems Inc. MEG/EEG
Operation and Technical Reference Manual & Omega 2000 Software User Manual,
Version 4.14

Table of Contents

INSTRUCTIONS

FLOW CHART

Section I

REVIEW, EDIT & PROCESS MEG DATA

	Page
I. Accessing Data	6
II. Filtering Data	8
III. Editing Data	9
A. Creating / Identifying ADC Markers using Threshold Detector	9
B. Creating / Identifying Stim Markers using DataEditor	12
C. AddMarker Command Line Program Options	14
D. Marking Eye Blinks & Other Artifact Using Threshold Detector	15
E. Classifying / Excluding Eye Blink Artifact using MATLAB & EEGLAB / ICA	16
F. Classifying / Excluding A Single Channel Using DataEditor	18
G. Classifying / Excluding An Entire Trial Using DataEditor	19
IV. Averaging Data	21
A. Averager Program	21
B. Averaging While Excluding Eye Blinks	23
C. averageDs Command Line Program	26

Section II

MRI DATA CONVERSION

I. Importing the MRI Images	27
II. Previewing & Specifying Parameters	29

Section III

INTEGRATING MEG & MRI

I. Co-registering Data Using Fudicials	31
II. Orthogonalizing the MRI	34
III. Creating a Head Shape	35
A. Accessing Data	35
B. Segmentation	35
1. Edge Enhancement	35
2. Surface Extraction / Erosion	35
3. Showing All Points	36
C. Thresholding	36
D. Best-Fit Sphere Model	38
E. Aligning / Matching Head Shape with Fudicials	39

Section IV

SOURCE MODELING /SAM ANALYSIS

	Page
I. Creating a SAM Volume (samCov)	41
A. Specifying Parameters	41
B. Opening the SAM Volume	45
C. Thresholding the SAM Volume	46
II. SAM Statistical Analysis Command Line Programs	49

Section V

OTHER ANALYSIS METHODS

I. AFNI (Analysis of Functional NeuroImages) for Group Analysis	
*Contact CIT / AFNI for technical support.	

Section VI

APPENDIX

File Paths and Name Conventions	52
---------------------------------	----

INDEX

Instructions

The MEG Data analysis process involves several steps consisting of various kinds of data manipulations that are required before the data can be analyzed. In addition, there are several different methods for performing these manipulations as well as several different kinds of analysis applications. It is not our intention to provide the user with a turn key method for performing data analysis rather we will provide a basic procedure and foundation for understanding the software and the processes that are currently available for performing MEG data analysis. The recommended procedure for analysis is outlined below.

The flowchart located on the next page is provided to guide users through the basic steps in the MEG data analysis process. The steps that do not have an asterisk are mandatory. The other steps are optional depending upon what kind of analysis is desired or preferred.

Basic procedure for MEG analysis:

1. **Filter data** by adjusting the processing parameters using DataEditor. This usually includes removing the DC offset, high pass filter, powerline.
2. **Inspect, edit, and process** each dataset using DataEditor to remove eye blinks and mark trials or channels as bad, if required.
3. **Generate markers** identifying the time of occurrence of events to be averaged or imaged by SAM using DataEditor.
4. **Average trials** using Averager.
5. **Import and create a MEG – formatted MRI file** from raw MRI slices using MRIConverter.
6. **Co-register the MEG and MRI coordinate systems** (designate the nasion and bilateral pre-auricular fiducial marks to agree with the positions of the MEG system head coils) using MRIViewer.
7. **Orthogonalize the head coordinates** (rotates or re-orientes the MRI Coordinate System with the MEG Head Coordinate System).
8. **Create a head shape file** containing only the boundary voxel coordinates using MRIViewer.
9. **Create a realistic (Best-Fit) head model** for each MEG dataset from the head shape file using localSpheres.
10. **Generate data covariance files** for chosen windows relative to the event markers using samCov.
11. **Generate SAM Volumes** using samSrc.
12. **Import and display color-coded “volumetric images”** produced by SAM software using MRIViewer or AFNI.

Data Analysis Flow Chart

Review Edit & Process MEG Data

I. ACCESSING RAW DATA

II. FILTERING DATA (PROCESSOR PARAMETERS)

III. EDITING DATA

Creating Markers / Identifying ADC Markers Using Threshold Detector

Creating Markers / Identifying Stimulus Markers Using Dataeditor

addMarker Command Line Options*

Marking / Removing Eye Blinks or Other Artifact Using Threshold Detector*

Excluding / Removing Eye Blink Artifact Using MATLAB / ICA*

Excluding a Single Channel Using DataEditor*

Excluding an Entire Trial Using DataEditor*

IV. AVERAGING DATA

Averager Program

Averaging While Excluding Eye Blinks*

averageDs Command Line Program*



MRI Data Conversion

(MRIConverter)

I. IMPORT MRI FILES

II. PREVIEWING & SPECIFYING PARAMETERS



Integrating MEG & MRI

(MRIViewer)

I. CO-REGISTERING DATA / SET FIDUCIALS

II. ORTHOGONALIZE MRI

III. CREATE HEAD SHAPE

Accessing Data

Segmentation

Thresholding

Create Best-Fit Sphere Head Model

Aligning / Matching Head Shape With Fiducials



Source Modeling / SAM Analysis

I. CREATING A SAM VOLUME (samCov)

Specifying Parameters

Opening the SAM Volume

Thresholding the SAM Volume

II. SAM STATISTICAL ANALYSIS COMMAND LINE PROGRAMS



Other Analysis Methods

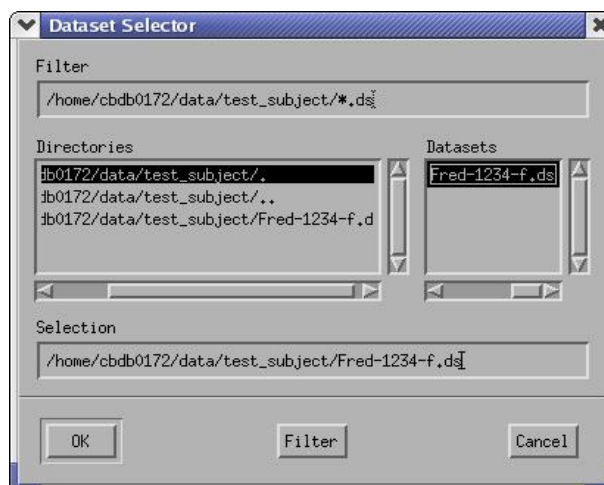
AFNI*

Review, Edit & Process MEG Data

All raw data should be processed, marked, edited and / or classified to remove environmental and / or physiological artifacts. In addition, data should be edited to mark trials, mark individual data points within a trial of data, mark different conditions or events, etc.

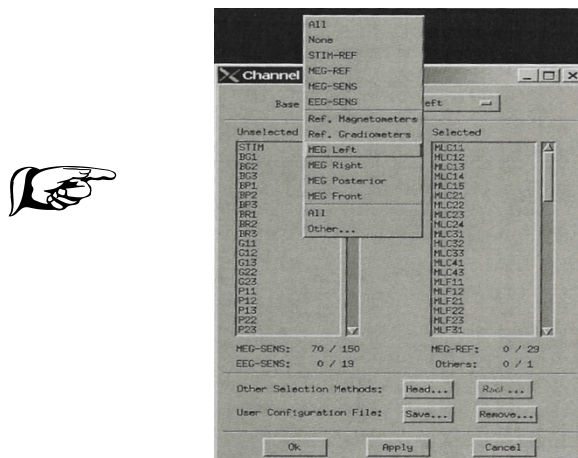
I. ACCESSING RAW DATA

1. Open a terminal window.
2. Type "**DataEditor**" <Enter>.
3. Select file from Dataset Selector > Directories.
4. Double click on the appropriate file under Directories.
5. Then click on the appropriate dataset under Datasets.
6. Select **OK**.




Dataset Selector

Note: Do not open a dataset as a directory.




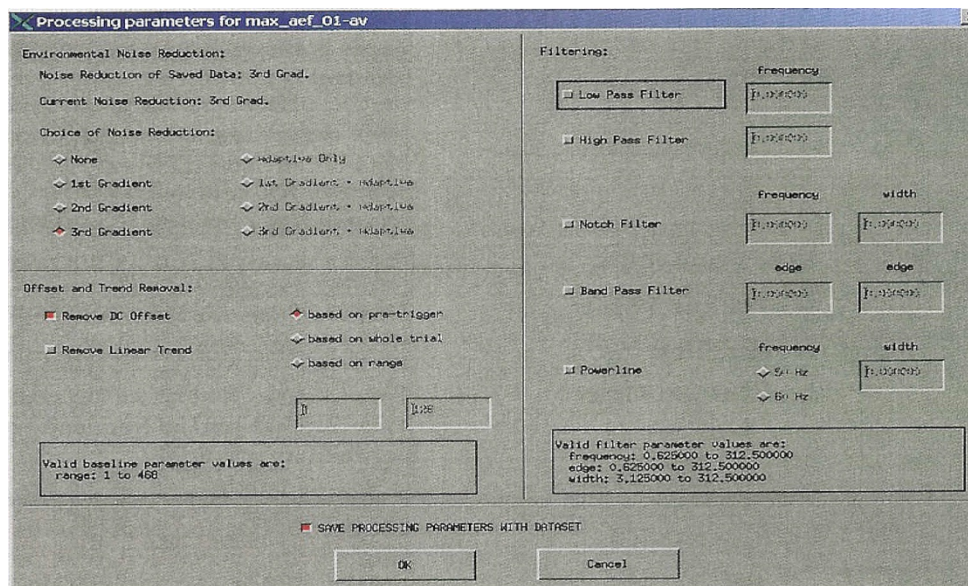
Channel Selector Dialog

7. Click on **Display > Channel Selector** (or Icon ).
8. Select channels to be displayed from the list or array by clicking on the channel in the unselected list to highlight.
9. Move channels to the **Selected** Menu by clicking on the arrow.
10. Click **OK**.
11. Adjust the amplitude and time scale using the Gain Control arrows located on the Control Toolbar which is at the bottom of the main window.

II. FILTERING DATA

Basic signal processing to filter data to remove environmental noise and low frequency artifact (e.g., breathing artifact).

1. Click on the **Analyse > Processing Parameters** (or Icon .
2. Select appropriate values for all parameters.



[Processing Parameters](#)

Processing Parameters	
Noise Reduction:	3 rd gradient; this is pre-selected if data was collected in this mode.
Offset and Trend Removal:	Select: Remove DC offset . Select: Based upon whole trial .
Low Pass Filter:	Generally not used at this stage.
High Pass Filter:	Valid range given by box at bottom of window (Can be used to remove breathing artifact or other low frequency activity that is not removed by 3 rd gradient.)
Band Pass Filter:	Generally not used at this stage.
Notch Filter:	Typically not necessary.
Powerline Filter:	60 Hz (Valid range for width given by box at bottom of window)


3. Click on **Save processing parameters with datasets**. (This is checked by default)
4. Click on **OK**.
5. Click on **File**.
6. Select **"Save dataset as"**. Enter new dataset name.
7. Click on **OK**.
8. If a popup window asks, "Do you want to load the setup file with this dataset?" select **NO**.

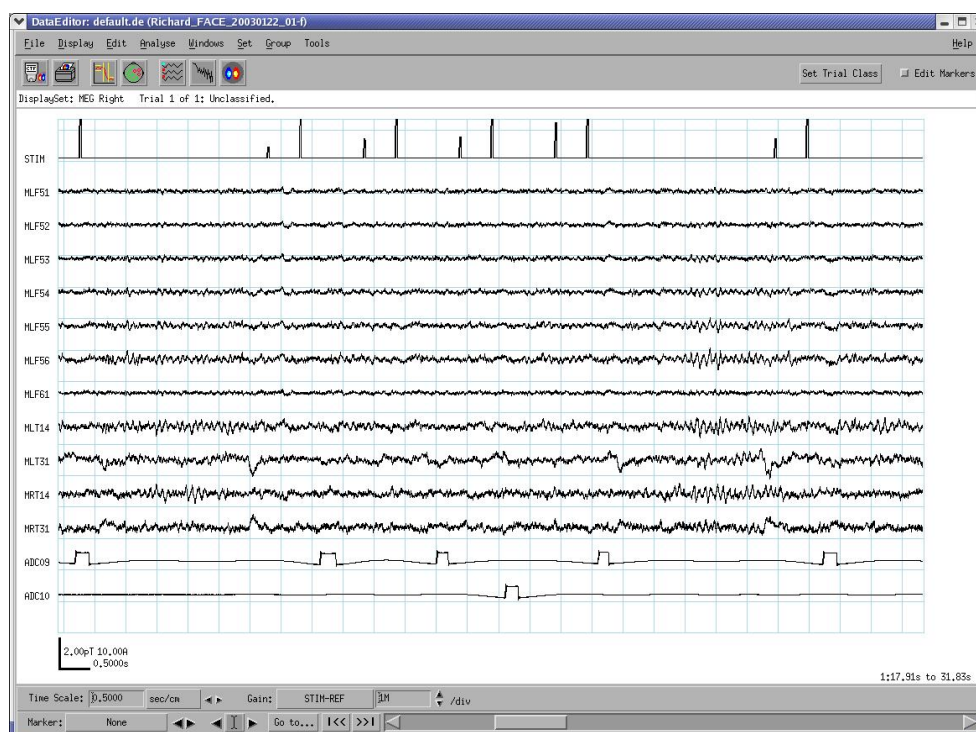
III. EDITING DATA

There are two types of markers: trigger markers (created during acquisition) and manual markers. Editing the data consists of creating identifying markers for the trigger markers (ADCs and stimulus). In addition the data should be edited to mark, exclude or classify various artifact, distractors or events of interest.

DataEditor, Threshold Detect program, independent component analysis (ICA) and various command line programs can be used to generate and edit markers identifying responses, stimulus markers, etc. and to mark individual data points within a trial of data. A channel, dataset, trial or specific event can be marked, excluded or classified using various methods. These methods are outlined below. It is up to the user to determine which method might be more useful.

A. CREATING / IDENTIFYING ADC MARKERS USING THRESHOLD DETECTOR:

1. Click on **Display > Channel Selector** (or Icon ).
2. Select STIM (from the top of the list) & ADCs channels (from the bottom). Move channels to the “Selected” box.
3. De-select MEG channels by highlight unnecessary channels in the “**Selected**” menu and move to the “Unselected” box.
4. Click on **Apply > OK**.



5. Click on the names of the ADC channels in the left margin.
This will highlight the channels in **RED**.

6. Click on **Analyse** Menu.
7. Click on **Threshold Detect** Program.
8. **Edit markers** button (top right of the DataEditor window) must be pressed to manipulate set markers.
9. Click on **Select Channels**. Enter **Threshold Detect** Parameters.

Threshold Detector

MEG: EEG/ADC:

Amplitude Threshold: 2p T 100u V

Derivative Threshold: 1p T/sec 100u V/sec

Dead Time: 100m sec

Save marker as: ☒ THRESHOLD

Classify trial as: ☒ BAD THRESHOLD

Channels: MLF11 MLF12 MLF21 MLF22 MLF23 MLF31 MLF33 MLF34 MLF43
MLF44 MLO21 MLO22 MLO31 MLO32 MLO33 MLO41 MLO42 MLO43 MLT11
MLT12 MLT21 MLT22 MLT23 MLT24 MLT25 MLT26 MLT31 MLT32 MLT33
MLT34 MLT35 MLT41 MLT42 MLT43 MLT44

☒ Scan All Trials ☐ Scan Current Trial Only

Apply Close

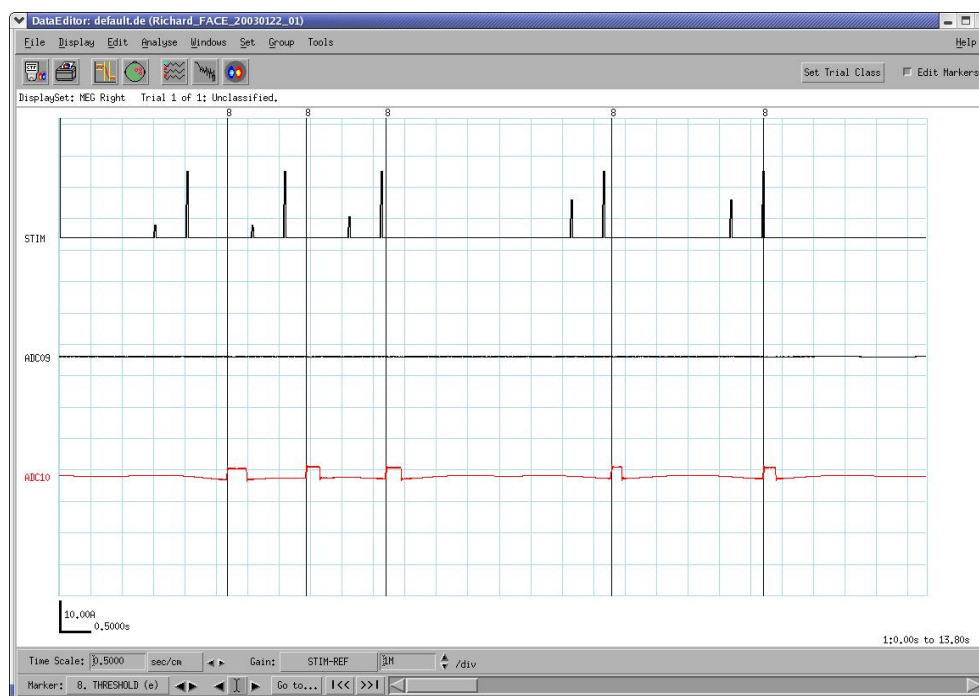
Threshold Detector

Threshold Detect	
EEG/ADC Amplitude Threshold (volts):	e.g. 1000u
EEG/ADC Derivative Threshold (volts/sec):	e.g. 500u
Dead Time (milliseconds):	e.g. 500-750m
Save Marker as:	Selected by user; marker name (e.g., Threshold)
Classify trial as:	Selected by user (e.g., Threshold)
<p><i>*These values, e.g., Dead Time, may change at the discretion of the user. Sporadically misplaced marks can be removed by clicking and then dragging them off the bottom of the screen.</i></p>	

10. Click on **Select Scan All Trials**.
11. Click on **Apply**.



Note: it may be easier to threshold detect ADC Channels before filtering.



12. General Information Popup will give # of detected events in # of trials.
If this is correct click **OK**.
 - a. If too many points are selected, re-adjust the Threshold Detect parameters.
13. **Close**.
14. Click on **Markers** at the bottom left corner of the window to choose the markers to display marker sets, which are highlighted in green (the color can be changed with **Edit > Marker Set**).

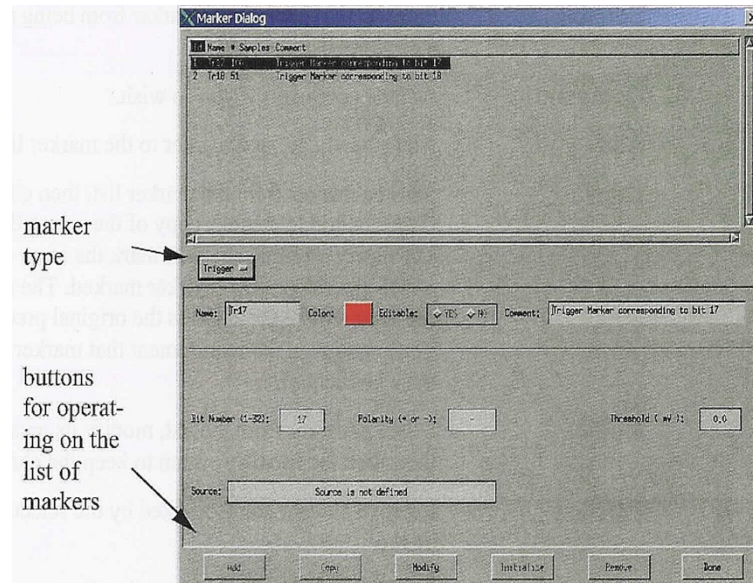


Note: Number Lock should be off in order to activate the Marker Button.

B. CREATING/IDENTIFYING STIMULUS MARKERS (TRIGGER MARKERS) USING DATAEDITOR:

*Note: To identify and edit a set of markers, the **Edit Markers** button in the top right of the screen must be pressed. This allows new markers to be added, removed, or modified.*

1. Click on the **Edit** menu.
2. Click on **Marker Sets** to display the Marker Dialog.

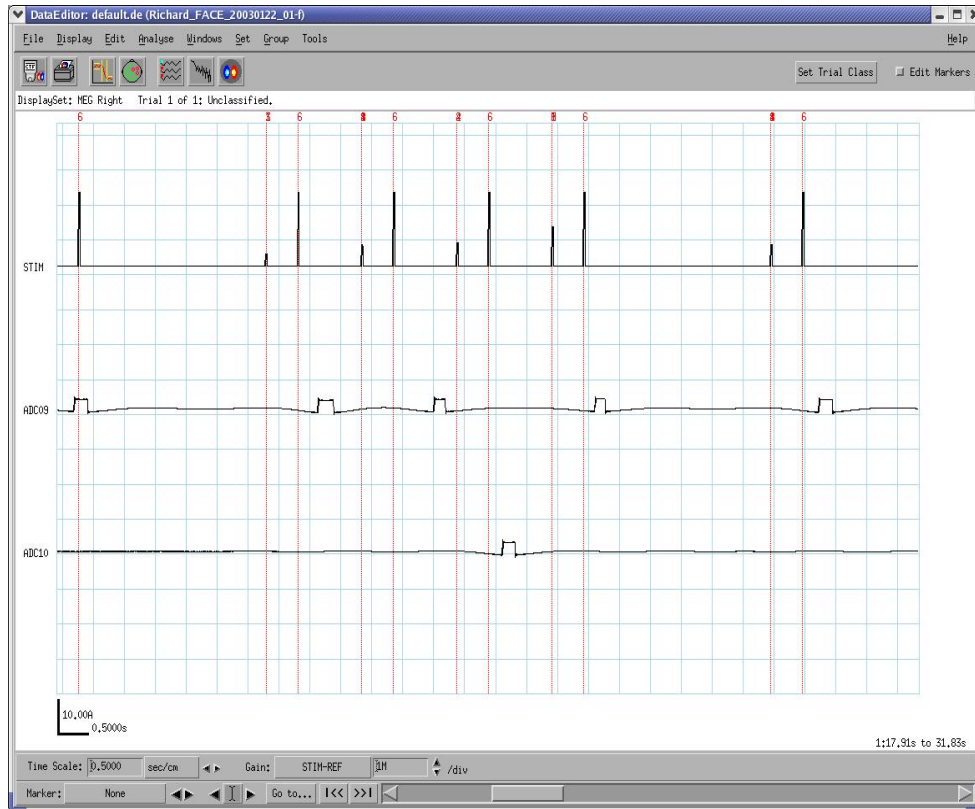


Marker Dialog

3. Click on the **Manual / Trigger** option and choose which type of markers you wish to view. Manual markers are made manually in DataEditor or in a command line program. Trigger markers are created during data acquisition and cannot be edited.
4. At the Editable option click on **YES**.
5. At the Color box select a color to distinguish between markers.
6. Click in the name box to type in the name of the marker.
7. Click on **ADD > Initialize**.
8. After identifying all triggers click on **DONE**.
9. Click on the **Display** menu.
10. Click on **Show>Markers**.



Note: Refer to example on the next page.



Note: Several cursor bars will appear on the screen depending upon the number of events recorded.

C. ADDMARKER COMMAND LINE PROGRAM OPTIONS:

For more complex options stimulus markers can be identified and/or made using the **addMarker** **–[option]** **<dataset.ds>** command line program.

The **addMarker Program** scans the stimulus channel of a dataset and marks samples that match a specified bit pattern (**–c** or **–q** option). It is also used to add markers from another marker file or user-created text file (**–a** or **–p** option).

These are some of the available options:

addMarker– Program		
<i>(Use addMarker –help to display a menu with a full range of options)</i>		
–n option:	Marker Name	Names the marker to create.
–c option:	Pattern	Scans the stimulus channel of a dataset and mark samples that match a specified bit pattern (–c <pattern>).
–q option:	Bit	Specifies a bit in the stimulus channel to match (–q <bit> <value>).
–t option:	Time Offset	Marks a specified time point in each trial (–t <time offset>).
–p option:	Text File	Adds markers from a text file (–p <text file>).
–a option:	Marker File	Copies markers from a dataset's marker position text file (–a <marker file>).
<small>*This program may also be used to create empty marker sets by specifying a marker name without specifying a condition pattern and/or a time offset.</small>		

1. Type **“addMarker –[option] <dataset>”** at the terminal window.

Example 1 – Creating marks from the bits of the Presentation stimulus code:

```
addMarker -q 17 1 -q 18 0 -n one someds-f.ds
addMarker -q 17 0 -q 18 1 -n two someds-f.ds
addMarker -q 17 1 -q 18 1 -n three someds-f.ds
```

Example 2 – Copying markers from a raw dataset to the filtered version:

```
addMarker -a someds.ds/MarkerFile.mrk someds-f.ds
```

Example 3 – Adding a marker from a user-generated text file:

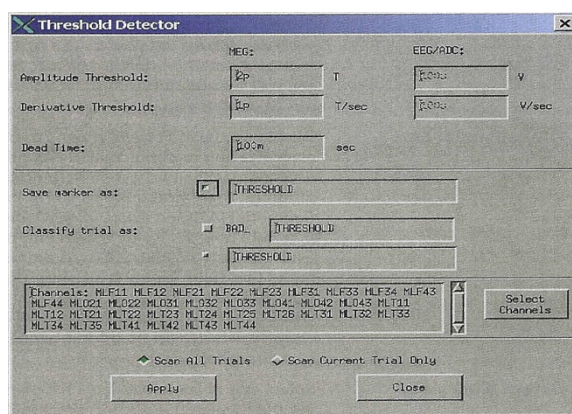
```
addMarker -p marker.txt -n stim1 someds-f.ds
```

Note: marker.txt has two columns: trial (starting from 0) and time (in seconds).

D. MARKING EYE BLINKS OR OTHER ARTIFACTS (PERFORM ON FILTERED DATA):

The Threshold Detect program can be used to mark eye blinks, other artifacts or to classify trials. Threshold Detect automatically marks data points and/or classifies trials based on the detection of a threshold event in the data. This program scans the specified channel(s) of a dataset and marks and/or classifies the samples where a specified threshold is exceeded.

1. Click on **Display**.
2. Click on **Channel Selector**.
3. Select MEG or EEG (that show artifacts to be removed, i.e., frontal temporal channels showing eye blinks) from the list.
4. Move to the "Selected" box.
5. Highlight unnecessary channels in the "**Selected**" menu.
6. Move to the "Unselected" box.
7. Click on **Apply > OK**.



[Threshold Detector](#)

8. Edit Markers (top right screen) must be pressed to manipulate set markers.
9. Click on the name of the channel in the left margin. This will highlight the channels in **RED**.
10. Click on **Analyse > Threshold Detect**.
11. Enter **Threshold Detector** Parameters.

Threshold Detector	
EEG/ADC Amplitude Threshold (volts):	Selected by user (depends on blink amplitude: MEG about 1pT)
EEG/ADC Derivative Threshold (volts/sec):	Selected by user (depends on blink amplitude: MEG about 1pT)
Dead Time (milliseconds):	500-750 msec; may change at the discretion of the user.
Save Marker as:	Selected by user (e.g., Blinks -for eye blinks; Artifact – for miscellaneous artifact, etc).
Classify trial as:	Bad

12. Click on **Scan All Trials**.
13. Click on **Apply > OK**.
14. Scan through data to make sure only eye blinks / specified artifacts were selected. Parameters may need re-adjustment.
15. Sporadically misplaced marks can be removed by clicking on them, dragging off the screen and then releasing.

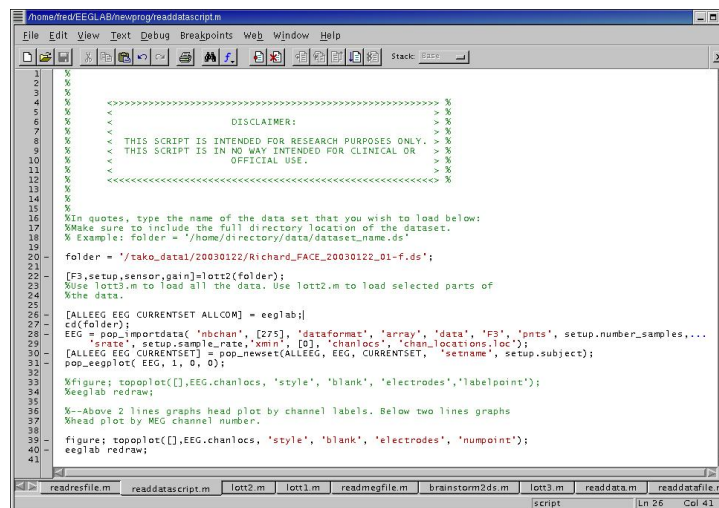
16. Close

E. EXCLUDING / REMOVING EYE BLINK ARTIFACTS USING MATLAB & EEGLAB/ ICA:

EEGLAB and a series of programs in MATLAB can be used to read data, convert it to an EEGLAB readable format, view and manipulate the data. After the data is loaded into EEGLAB using `readdatascript.m`, an ICA algorithm can be used to remove eye blink artifact and to redisplay the data. `readdatascript.m` is a MATLAB script that accesses a series of other programs to read an inputted dataset's header information, gains/offsets, and raw data. The script then imports the data into EEGLAB to show plots of the channels and the data itself.

DISCLAIMER: `readdatascript.m` and all other programs associated with it are intended for research purposes only. In no way are these programs intended to be used clinically or officially. Other programs include `readmarkerfile.m`, `readmegfile.m`, `readresfile.m`, `readdatafile.m`, `lott1.m`, `lott2.m`, `lott3.m`, `readdata.m`, `readdatascript.m`.

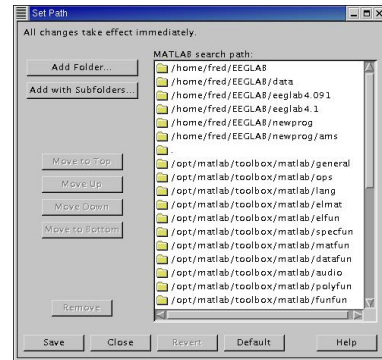
1. Open **MATLAB**.
2. Click on **File>Open**.
3. Click on the appropriate directory and select **readdatascript.m**.
4. Click **OK**.




```
1 %  
2 %  
3 %  
4 %  
5 %  
6 %  
7 %  
8 %  
9 %  
10 %  
11 %  
12 %  
13 %  
14 %  
15 %  
16 %  
17 %In quotes, type the name of the data set that you wish to load below:  
18 %Make sure to include the full directory location of the dataset.  
19 % Example: folder = '/home/directory/data/dataset_name.ds'  
20 folder = '/tako_data/20030122/Richard_FACE_20030122_01-f.ds';  
21  
22 [F3,setup,sensor,gain]=lott2(folder);  
23 %Use lott3.m to load all the data. Use lott2.m to load selected parts of  
24 %the data.  
25  
26 [ALLEEG EEG CURRENTSET ALLCOM] = eeglabj;  
27 cd(folder);  
28 EEG = pop_importdata('nbchan',[275], 'dataformat', 'array', 'data', 'F3', 'pnts', setup.number_samples,...  
29 'rate', setup.sample_rate,'xmn', [0], 'chanlocs', 'chan_locations.loc');  
30 [ALLEEG EEG CURRENTSET] = pop_newset(ALLEEG, EEG, CURRENTSET, 'setname', setup.subject);  
31 pop_eegplot( EEG, 1, 0);  
32  
33 %figure; topoplot([],EEG.chanlocs, 'style', 'blank', 'electrodes','labelpoint');  
34 %eeglab redraw;  
35  
36 %--Above 2 lines graphs head plot by channel labels. Below two lines graphs  
37 %head plot by MEG channel number.  
38  
39 figure; topoplot([],EEG.chanlocs, 'style', 'blank', 'electrodes', 'numpoint');  
40 eeglab redraw;  
41
```

5. The path to `readdatascript.m` may not be saved in MATLAB, in which case the script will not run. If this is the case, complete the following steps:

- a. In the main MATLAB window, click **File>Set Path**.
- b. Click on the box that reads **Add with Subfolders...**
- c. Click on the appropriate directory, or the folder where readdatascript.m is saved. Click **OK**.
- d. Click **Save** and then **Close**.



6. Return to the **readdatascript.m** window.
7. On line 5, where it says “folder = “, type the path and dataset that you wish to view in single quotes. Example: **folder = '/tako_data1/20030122/Richard_FACE_20030122_03-f.ds'**
8. To save, click **File>Save**, or the Save Icon.
9. Click on the **Run Icon** , or on **Debug>Run**.
10. Return to the main MATLAB window to watch the script carry out its function. The data and channel locations should be plotted automatically.
11. Now use EEGLAB to remove the eyeblinks. Go to the new EEGLAB main window.
12. Click **Tools>Run ICA**.



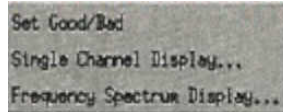
Note: ICA filters the data by finding pieces of data that seem uncorrelated with the rest of the data as a whole and removes that portion of the data.

13. In the **Run ICA Decomposition** window, type in “**runica**” in the window labeled **ICA Algorithm to use**. The **Command Line Options** window can be left blank.
14. Click **OK**.
15. If the ICA Filtered dataset is not selected when the process is done, click **Datasets > ICA Filtered**.
16. To remove the marked components, click on **Tools > Remove Components**.
17. When EEGLAB asks, “Do you agree with the projection?”, click **Yes**.

F. CLASSIFYING/EXCLUDING A SINGLE CHANNEL USING DATAEDITOR:

If a channel is particularly noisy it can be marked as bad. This will classify the channel across all trials and by default the channel will be removed from averaging, etc.

1. In DataEditor select a channel by **clicking the channel's name** in the left margin with the left mouse button. The channel will be highlighted in **RED**.
2. Right click on the **channel's name** to bring up the pop up window to classify it as good or bad.
3. Set the status of the selected channel to either **good** or **bad**. *Setting a channel as "bad" will remove it from the DataEditor screen.



4. To display the channels that are classified as bad, select **Display > Show > Bad**
5. **Channels** menu options.

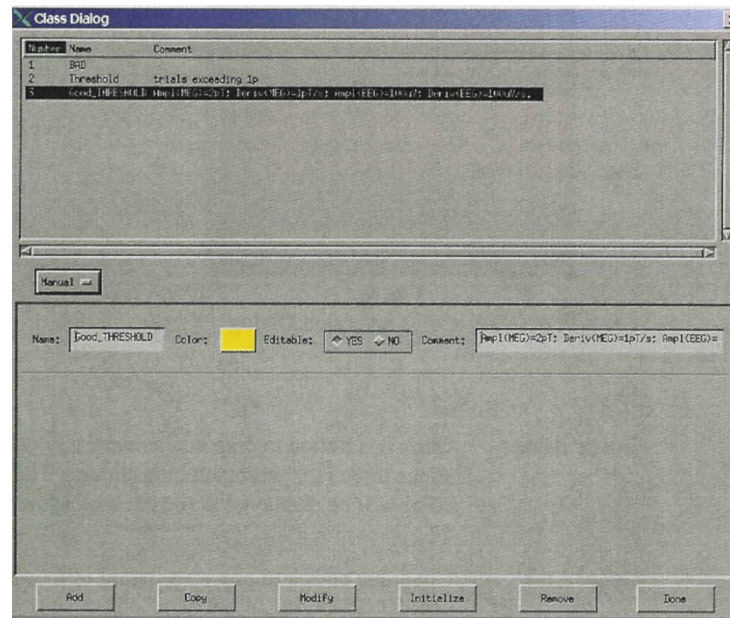
G. CLASSIFYING / EXCLUDING AN ENTIRE TRIAL USING DATAEDITOR:

Use to classify an entire trial. Trials can be classified as bad for the purpose of exclusion in averaging. This method can be used when a subject moved extensively, a subject coughed throughout the trial, you forgot to turn the stimulus on for the trial, etc. Trials can also be classified to identify them for other purposes, i.e., control trials, distractors, etc.

Once a trial classification is created, whole trials can be classified using the **Set Trial Class button in the upper left hand control bar.*

CREATING / MODIFYING A CLASSIFICATION:

1. In **DataEditor** scan through dataset.
2. Click on **Edit > Trial Classification**. The **Class Dialog** will display.



Class Dialog

3. Enter a name classification in the **Name** option (i.e., bad, average, delete, subject moved, etc.).
4. Click **Add**.
5. Click on the numbered marker, at the top of the class dialog, that you wish to classify.
6. Click on **Color**. Select a color to distinguish the trial from other trials.
7. Click **Modify** after making any changes.
8. At the **Editable** option click on **YES**.
9. Under **Comment** include a description of the event or trial.
10. Click on **DONE**.

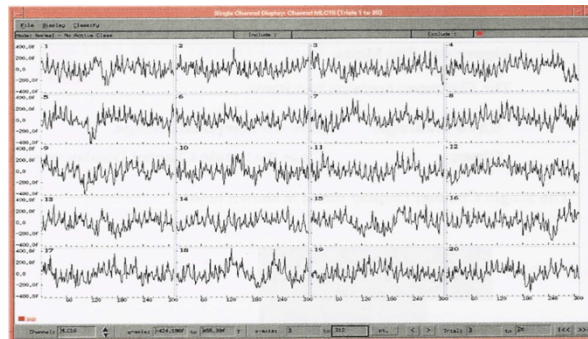


Note: Continued on the next page.

CLASSIFYING TRIALS (USING SINGLE CHANNEL DISPLAY):

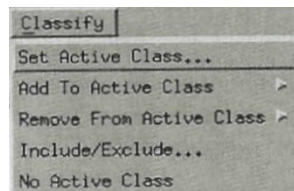
The Single Channel Display option can also be used to select one or more trials for classification. *Note: Although shown in single channel display, trials are classified (and excluded) across all channels in the dataset.*

1. Designate the channel by clicking on it in the left margin of the DataEditor window. The channel will be highlighted in **RED**.
2. Right click on the channel then click on **Single Channel Display**.



Single Channel Display

3. Select **Classify > Set Active Class**.



4. Select one of the classes listed in the **Select Class Dialog**. This will make the selected classification active.
5. Click on **Apply > OK**. The menu bar at the top of the window will display the:
 - a. mode,
 - b. name, and
 - c. color of the active class.
6. To add additional trials to the active class, click on the trial and it will change color from black to the class color.
7. To remove a trial from the class, click on the colored trial and the trial will change to black.
8. The classification of each trial is shown on the menu bar as you scan through the dataset. Use the control bar located at the bottom of the window to scan through the trials in the dataset. The control bar can be used to make other adjustments as well.
9. Click on **File > Close**.



Single Channel Display Control Bar

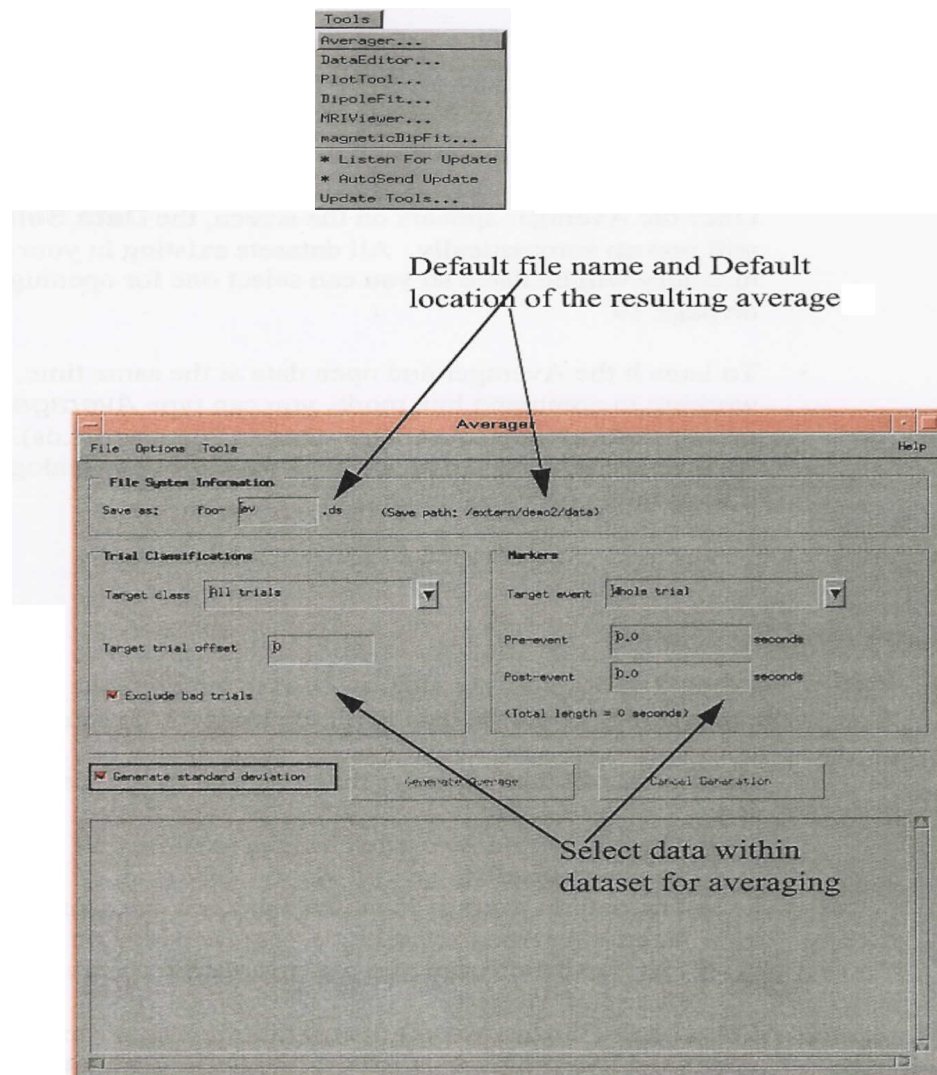
IV. AVERAGING DATA

The **average program** – is a program used for basic averaging which computes and displays an average of the data in a single dataset. The averaged data is stored in a new dataset. Before generating the average, specify what trials in the dataset to use (the default is all trials) and what events in the trials to use (the default is to use the whole trial).

This program requires an even number of events around which it will generate an average. If the source dataset has five trials and they are averaged as outlined below, only the first four trials will be used to generate the averaged dataset.

A. AVERAGER PROGRAM:

1. At the terminal window type **Averager <Enter>** or Click on **Tools > Averager** (in DataEditor).



2. The “Save Changes?” dialog will pop up automatically if changes have been made. Click on **Save**.
3. The Averager’s File System Information dialog displays.
 - In the **Save as** section check to make sure the correct dataset is entered.
 - In the Markers section select:

Target event:	Specify a marker, i.e., whole trial, Tr1 (short for trigger 1’s marker), 0-back, etc.
Pre event:	Specify a data range before the marker occurrence in seconds; user determined (positive value).
Post event:	Specify a data range after the marker occurrence in seconds.
Exclude bad trials:	Generally Selected; User determines.
Generate Standard Deviation	Generally Selected: User determines.

- Click on the **Generate Average** button to start the average program.
4. The General Information screen will pop up indicating that a new dataset has been created. Click **OK**.
 5. Use an overlay to compare the averaged dataset with the source data. Generally, one looks at an overlay with two –av datasets averaged around markers in order to see the differences.
 - Click on **Tools > DataEditor** menu
 - Select **File > Open Overlay**.
 - *The gray traces depict the raw data.

B. AVERAGING WHILE EXCLUDING EYE BLINKS:

Step I: DataEditor: *Marking eye blinks in the dataset. Refer to Section III.D.*

1. In a terminal window, type **DataEditor**.
2. Click on **File > Open**.
3. Select the appropriate directory and filtered dataset.



Note: *the dataset opened must have been filtered.*

4. Click on **Display > Channel Selector**.
5. Select the channels you want to view in the unselected list and move them to the selected list by clicking the black arrow. Channels that show the largest eye blink amplitudes tend to be MLT14, MLT31, MRT14, and MRT31.
6. Click **Apply** and then **Ok**.
7. Click on **Analyze > Threshold Detect**.
8. In the left margin, select the channel that best displays eye blink artifact.
9. Select **Scan All Trials**.
10. Save the marker as "blinks," or some other name to be used again later.

The threshold values will vary for each subject because eye blink amplitude varies. Some examples of threshold values for eye blinks are:

Amplitude Threshold	~0.7pT
Derivative Threshold	~1.0pT
Dead Time	~500ms

11. Click **Apply**.

Step II: addMarker: *Creating markers in data to average around.*

1. Open a new terminal window and type "**addMarker -help**" for help information. Refer back to the addMarker section of the manual for more information.
2. In the command line be sure to name your new marker and refer to the correct dataset.

An example would be:

addMarker -n Marker -q 17 1 -q 18 0 -q 19 0 Richard_FACE_20030122_01-f.ds

3. Go back to DataEditor and reopen the previously created dataset with the eye blink markers during Step I.

Step III: averageDs: Averages data around the markers created in Step II while excluding data around the eye blink markers created in Step I.

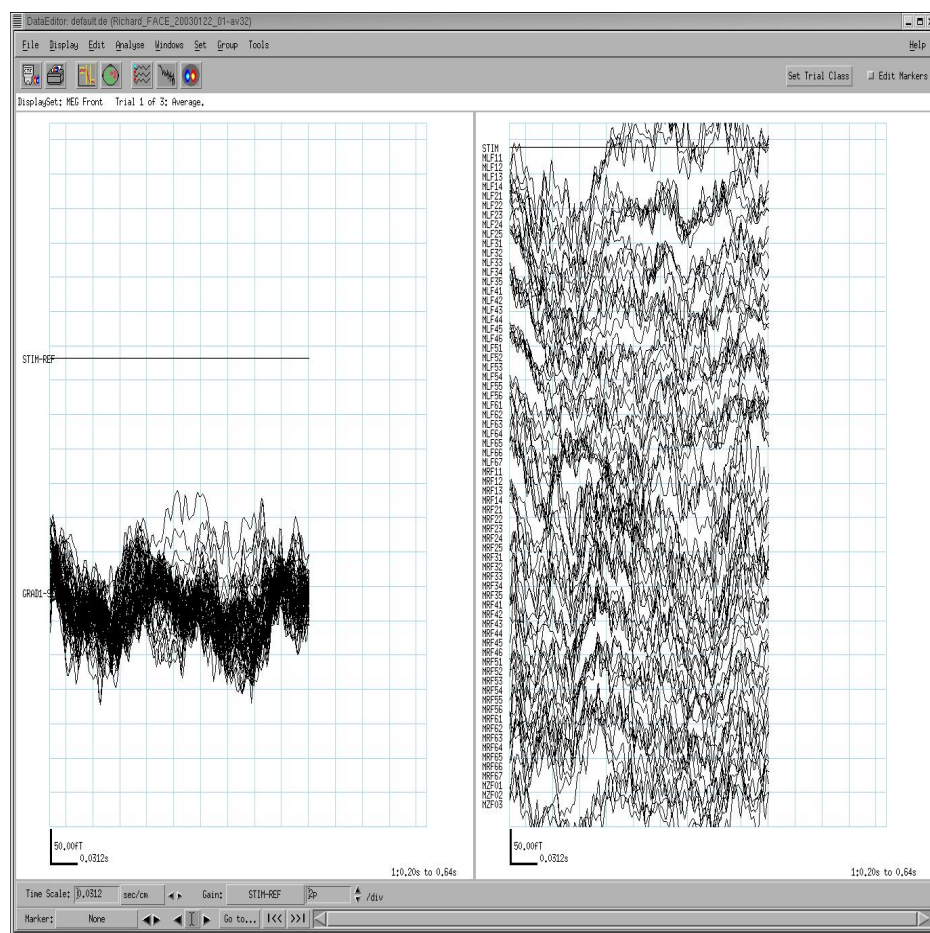
1. Open a terminal window and type “**averageDs –help**” for help information. Refer to the averageDs section of the manual for more information.
2. Specify a marker to average using the ‘-marker’ and ‘-time’ options.
3. Use the ‘-excludeEvent’ option to specify the blink marker and a time interval. Remember that time intervals are relative to the *target* marker. Thus the interval must include enough time at the beginning for both the “pre-target” window and the “post-blink” window, and enough time at the end for both the “post-target” and “pre-blink” windows, to ensure exclusion of the entire blink.
4. Also include the name of the dataset these markers are in and a filename for a new, averaged dataset (-av).

Example: Assume that a “blinks” marker marks the beginning of a blink, and that the blink extends from .2 seconds before the marker until .4 seconds after it. Then to average around the “stim” target marker using a window of -.2 to .5 seconds, type:
averageDs -marker stim -time -.2 .5 -excludeEvent blinks -0.6 0.7 (continue ↵)
somed-f.ds someds-av.ds
where -.6 is -.2 minus .4 (to exclude the post-blink), and .7 is .5 plus .2 (to exclude the pre-blink).

5. Go back to DataEditor and open the new, averaged dataset that you have just created (*somed-av.ds* in this case).
6. Use the **Channel Selector** to select the averaged channels to view.
7. Click on **Display > Layout > Overlay Channel Types** to display averaged data.



Note: Refer to example on the next page.



Averaged Data

C. AVERAGEDS COMMAND LINE PROGRAM

averageDs – is a command line program which is designed for more specific and/or more complex averaging, i.e., to average only certain trials within a run or to include or exclude certain events (eye blinks) or if you want to average a specific length of time.

1. At the terminal window type **averageDs <Enter>**.
2. Type: **“averageDs –help”** command to list possible options for averaging.

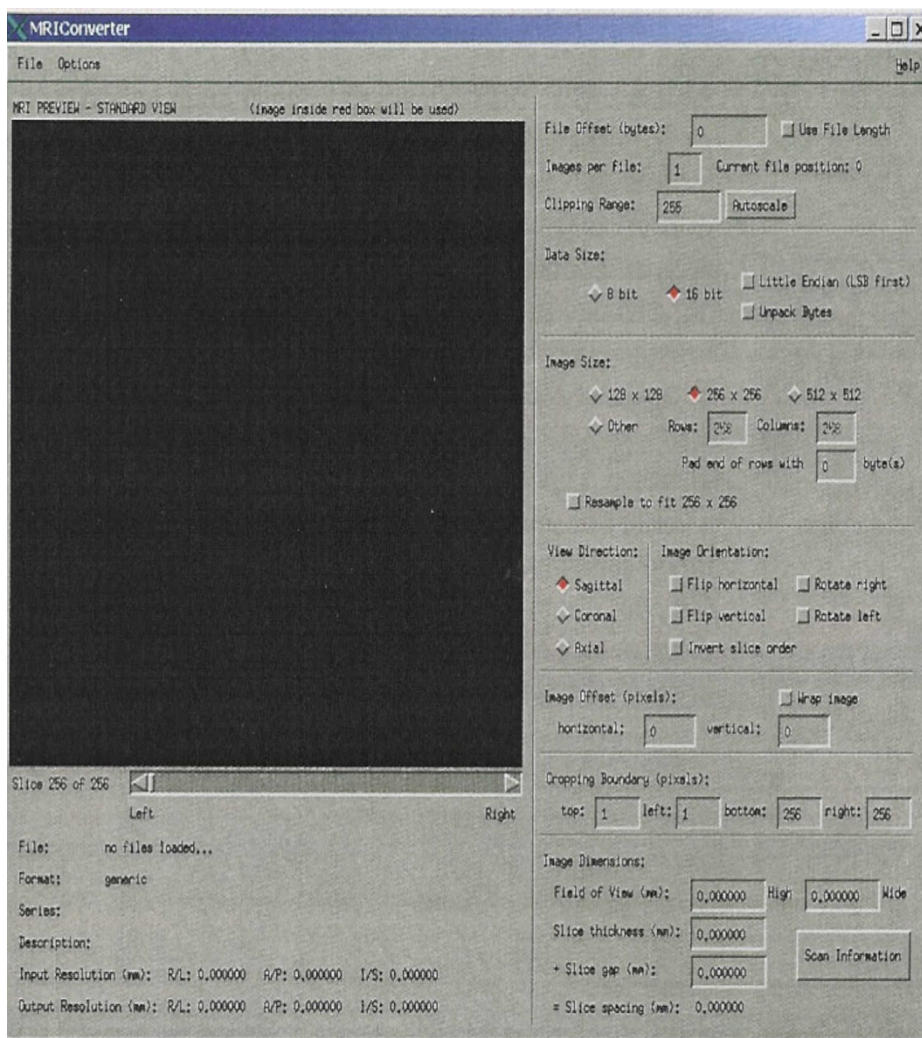
MRI Data Conversion

The MRIConverter program is a utility that allows the importation of Magnetic Resonance Imaging (MRI) data to produce an isotropic volumetric image file, which can be co-registered with the head coordinate system produced by the MEG/EEG system. This allows the importation of equivalent current dipole (ECD) or other types of source images for superimposition onto individual MRI slices.

I. IMPORTING MRI FILES:

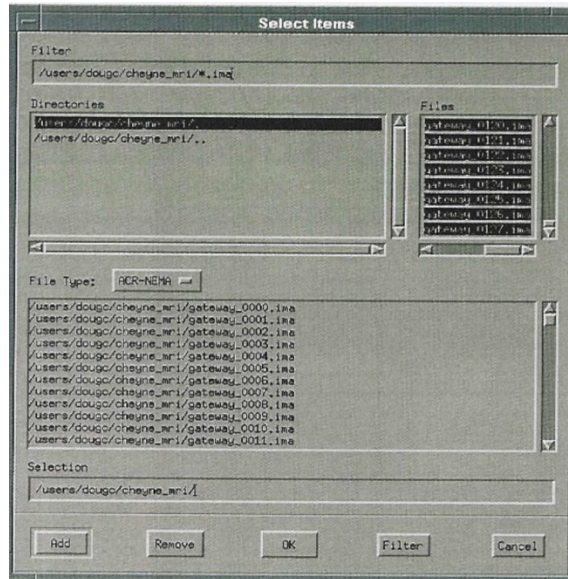
In order to import MRI data for the purpose of creating a volumetric image file, it is necessary to read in the original MRI data into memory. In most cases, this data will consist of a large number of individual files, each containing one MRI slice in the original scanning orientation.

1. Open a terminal window.
2. Type: **MRIConverter** <Enter>.



[MRI Converter](#)

3. Click on **File Menu > Open Files** (Raw Data).
4. Double click on the appropriate directory at the **Select Items** dialog.
5. Select all the files needed to create the MRI by clicking on the first file (on the right side) and dragging the mouse down.
 - * Files must be entered into the list in the same order as they were originally scanned.
6. Click **Add**.
7. Select MRI format (i.e., ACR-NEMA, DICOM 3, Siemens SPI, and Generic) from the **File Type** selector dialog. If you do not select an MRI format, you will be asked, "Couldn't determine filetype. Read as generic?" Click **Yes**.
8. Select **OK** to proceed.



9. A warning will display if there are missing values. Click **Ok**.

II. PREVIEWING MRI DATA & SPECIFYING INPUT PARAMETERS:

Once the file list selection is complete the **MRI Converter Dialog** will be displayed. If the files were imported in a pre-defined format (e.g., ACR-NEMA) many of the file input parameters will have been automatically set and the correct image of the MRI slices in the MRI Preview window will be displayed. However, some parameters will need to be changed. In many pre-defined file types the file headers may be incomplete or contain incorrect information. Therefore it is recommended that all file input parameters are checked for their correctness before creating the volumetric image file.

Although all of the necessary input parameters can be determined from file headers in pre-defined file formats, the relationship between the directions of scanning and the order in which the files have been entered cannot always be determined. This may be difficult in the case of input files scanned in the sagittal direction since an inversion of the image left-to-right cannot be discerned from simply viewing the image. In this case, independent information such as photographic output from the MRI scanner may be necessary to determine the correct order of input. If you are having trouble reading the header information, you can download MIPAV, a java application that reads headers of almost all MRI formats. Go to <http://mipav.cit.nih.gov> for more information.



[MRI Converter](#)

Data Size:	16 Bits (pre-set)
Image Size:	256 X 256 (pre-set)
View Direction:	*depends on orientation of scan.
Image Dimensions:	*comes from header information. Usually: 240 High; 240 Wide.
Field of View:	Depends upon scan.
Slice Thickness:	MRI slice thickness, which is found in header information. Typically ~1.5mm
Image Orientation:	1) Left should be on Left; if not flip horizontally. 2) Make sure when moving the arrow to view the slices that superior moves the scan in the superior direction if not invert the slice order.

1. Change to the above indicated settings.
2. Click on **File > Save as**.
3. If the "Are you sure?" dialog will pop up automatically, click **Yes**.
4. If asked, "Create volumetric image file using current settings?" click **Yes**.
5. Enter the file name.
6. Save as **8 bit**.
7. Click on **OK**.

Integrating MEG & MRI

MRViewer creates and uses CTF's format for reading and viewing volumetric MR images. The MR image may be viewed in either neurological or radiological orientations. The MRI Viewer is used to co-register the MRI coordinate system with the MEG Head Coordinate System (designate MRI fiducial marks so that they agree with the positions of the MEG system head coils). It is also used to orthogonalize MRI Images (rotate and re-slice the MR image in order to align the MRI axes with those of the MEG Head Coordinate System such that the X axis is aligned to the coronal direction); to create a head shape file containing only the boundary voxel coordinates and invokes localSpheres to create a realistic head model.

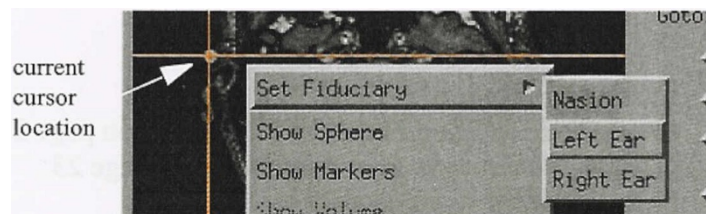
I. CO-REGISTRATION USING FIDUCIAL LANDMARKS:

After creating a CTF format MRI file from the original data files, it is necessary to define the location of the fiducial landmarks used in the head localization software for correct "co-registration" between the MEG head coordinate system and the MRI coordinate system.

1. From a terminal window type **"MRViewer"**<Enter>
(or from the Tools menu select MRViewer).
2. Click on **File > Open MRI File...**



3. Double click on the appropriate directory at the **Select Items** dialog.
4. Double click on the file in the **File** section.
5. Click on **OK**.
6. Position the cursor over the position marked on the digital pictures matching the desired location. Compare the MRI fiducial placements to the dots (marks) in the digital pictures for accuracy.
7. Right mouse click to set the fiducial mark.

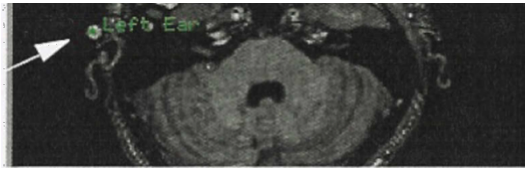


8. Click on **Set Fiduciary**. Select Nasion, Left Ear or Right Ear.



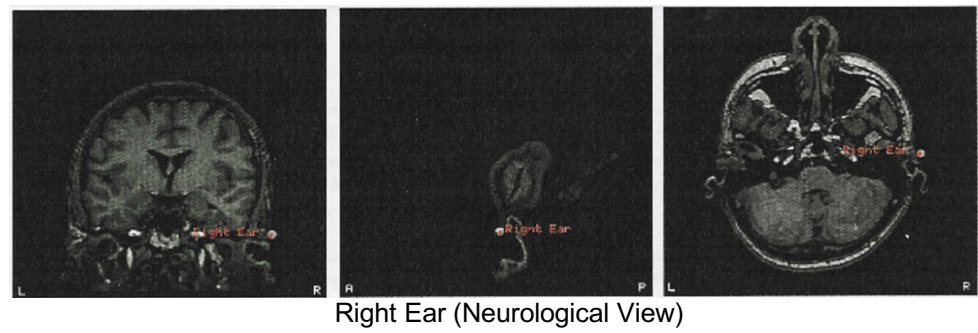
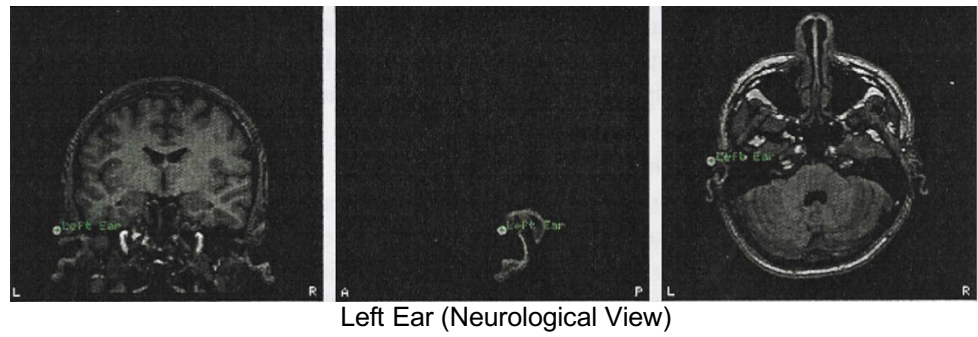
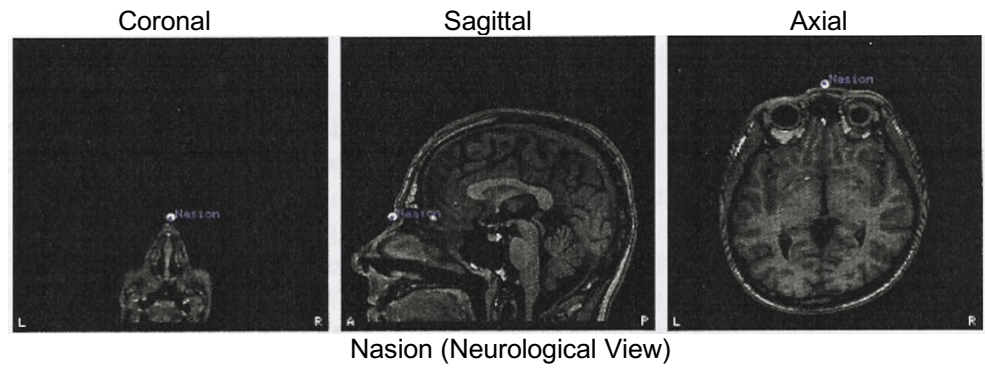
Note: CTF does say "fiduciary" but the correct term is fiducial.

9. After releasing the mouse button the resulting location of the Fiducial point is marked by a color coded label.



Note: Clicking lock view to cursor helps to see the same crosshair location in all 3 views.

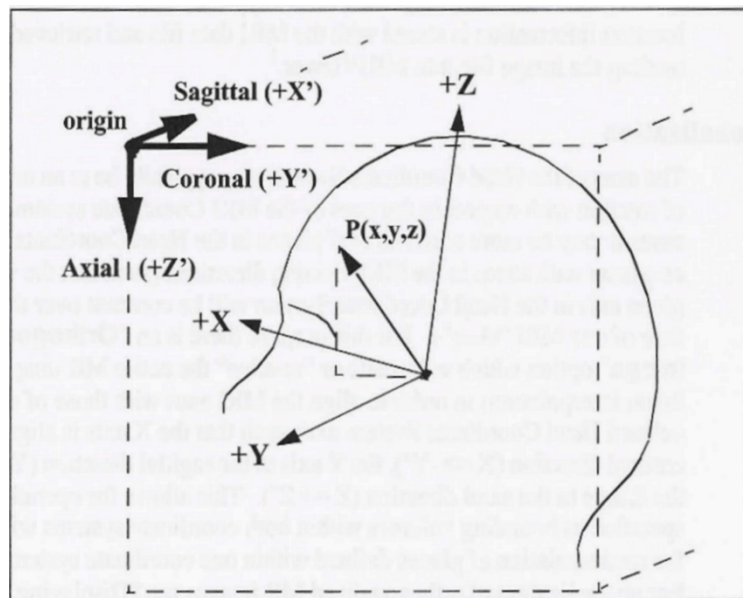
10. This procedure must be done for all three coil locations.



11. Click on **File> Save Changes** option.

II. ORTHOGONALIZATION:

The axes of the MEG Head Coordinate System may be at an arbitrary angle of rotation with respect to the axes of the MRI Coordinate system. The orthogonalization option will allow for the rotation or “re-slicing” of the entire MR image in order to align the MRI axes with those of the Head Coordinate System axes such that the X axis is aligned to the coronal direction, the Y axis to the Sagittal direction and the Z axis to the axial direction. This allows for operations to be specified as bounding volumes within both coordinate systems without need for re-interpolation of planes defined within one coordinate system to another. The fiducials must be defined before starting Orthogonalization.



The MRI Coordinate System's relationship to the MEG Head Coordinate System.



Note: Orthogonalization of the MR image is necessary for viewing MR images in AFNI.

1. From a terminal window type **MRViewer** <Enter>
(or from the Tools menu select MRViewer)
2. Click on **Options > Image Orientation**.

Options:

Radiological	(left on right)
Neurological	(left on left)

3. Select: **Neurological view**.
4. Click on **Options > Orthogonalize Image**.
5. At the Popup Question menu “**Make image orthogonal to head frame?**” click **YES**.
6. Save the MRI image as: *name-O*.
7. Exit and Restart MRViewer.

III. CREATE HEAD SHAPE:

The Head Shape window in MRIVIEWer is used to extract a list of head shape points. Although you can visually see the bounding surface of the MR image, its location cannot be described without first performing some sort of surface extraction. The bounding surface is computed by thresholding the gray scale values of the image's pixels. Pixels are set to black or white according to the threshold value. The resulting black and white image is passed through an erosion algorithm. This algorithm extracts the bounding surface points relative to the fiducial markers. A second pass through the erosion algorithm further removes extraneous points. Before thresholding, the original gray scale image may optionally pass through an edge enhancement algorithm first. This may improve the resulting thresholded image.

A. Accessing Data

1. From a terminal window, type **MRIVIEWer** <return> (or from the Tools menu select MRIVIEWer).
2. Click on **File > Open**.
3. Double click on the appropriate Directory.
4. Then double click on the MRI image under File.
5. Click **OK**.

B. Segmentation

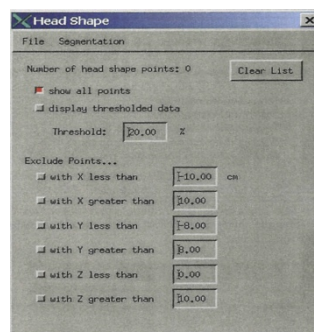
The Segmentation consists of extracting surface points to remove extraneous points and transforming the MRI for edge enhancement.

1. **Edge Enhancement (Optional):** – To sharpen the edges of the MRI. This should be performed before completing surface extraction.

- Click on **Segmentation**.
- Click on **Transform Data**.
- Click on the **Sobel Edge Enhancement**.
- Click **Close > YES** to restore image.

2. **Surface Extraction / Erosion** – Extracts the bounding surface points relative to the fiducial markers.

- Under MRI Viewer Menu click **Options > Head Shape**.
- Select **Segmentation** from the Head Shape Dialog window.



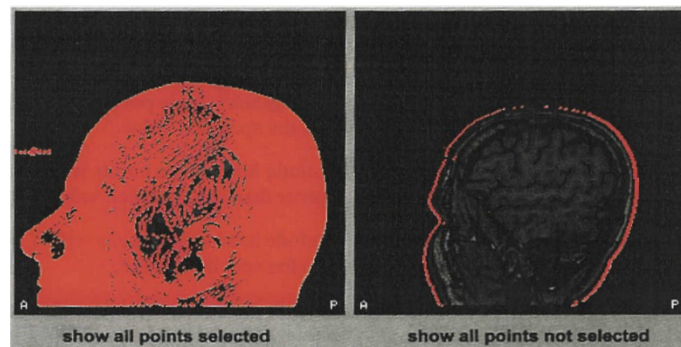
[Head Shape Dialog](#)

—Click on **Extract Surface**.

—Select:

- One-pass erosion – to generate the head shape points by performing a one-pass erosion on the selected MRI voxels to remove extraneous bounding surface points.
- Two-pass erosion – to generate the head shape points by performing a two-pass erosion on the selected MRI voxels to further remove extraneous bounding surface points.

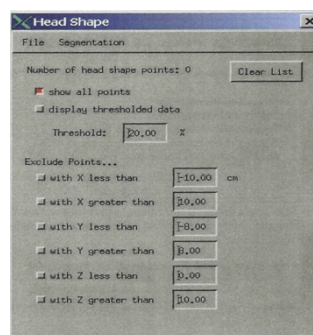
3. From the **Options > Head Shape** window, click on **Show all points** to display all head shape points onto the three MRI slices in the main window.



C. Thresholding

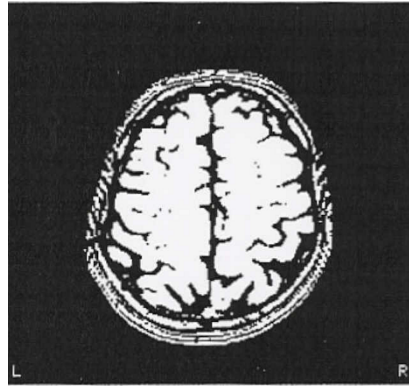
Used to exclude points in certain axes to limit the extracted surface to the desired brain area.

1. From the **Options > Head Shape Dialog** window,



[Head Shape Dialog](#)

2. Click on **Display thresholded data** to display the black and white image of the thresholded MRI in the main window.
3. Click on **Threshold %** to reduce the threshold value; this will increase the head shape points.



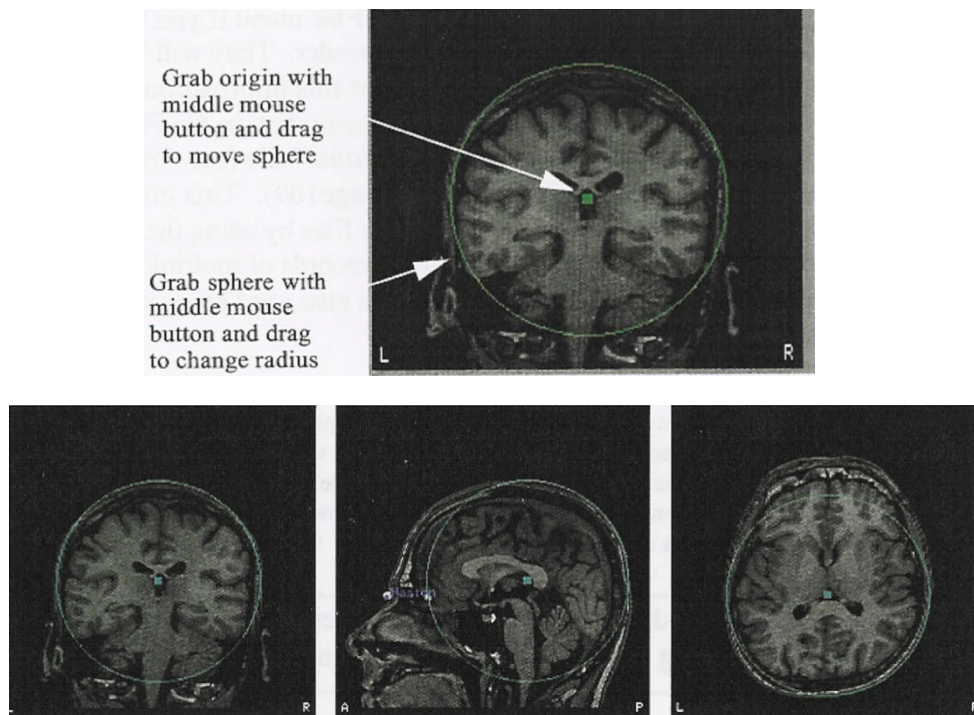
4. Use the **Exclude Points** options to limit extracted surface to the desired brain area.
5. Axes should be adjusted to best fit the individual subject.
6. Click **File > Save Head Shape**.

D. **Best-fit Sphere Head Model**

A spherical conductor approximating the inner skull surface is the model used in CTF's dipole source modeling programs. The spherical conductor is assumed to have concentrically uniform conductivities. The location and radius of this volume conductor is called the best-fit sphere. The best-fit sphere is the sphere which best approximates the subject's head shape. Using the MRI as a guide a "best-fit" sphere is created based on the anatomy of the patient's head for the purpose of source modeling routines, i.e., dipole fit or multiple local spheres and for SAM analysis. For EEG dipole modeling, additional spherical shells describing the various layers of the skull and brain may also be included.

Note: The head shape file must be open to make the best-fit sphere.

1. In MRI Viewer, click on **Fit to Headshape**.
 - if you are viewing a slice, which does not intersect the sphere origin, you will see the sphere surface that intersects that slice and no origin marker appears in the image.
2. To manually position the sphere:
 - When the sphere origin is visible, use the mouse button to grab the origin (position the mouse cursor over the origin and press the middle mouse button).
3. To adjust the sphere radius:
 - Grab the sphere's outline and drag.



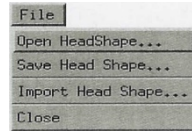
Coronal, Sagittal and Axial view showing the center of the best-fit sphere.

4. Click **File > Save Changes** if you want to save these as the default parameters in the MRI file.

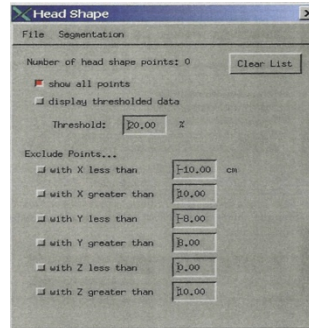
E. Aligning / Matching Head Shape with Fiducials

Open HeadShape:

1. Click on **File > Open HeadShape** to load the head shape data into the MRViewer.



File Menu



[Head Shape Dialog](#)

2. If the fiducial points of the head shape data are not the same as the currently specified fiducial points, the following question is displayed:

“Update fiduciary points to match those saved with this head shape or re-align points?”

- Click **Update** – to replace the current fiducial points with those from the head shape data file.
- Click **Re-align** – to keep the current fiducial points and realign the new head data points to the current head coordinate system.

3. If head shape data has already been loaded or computed, the following question dialog is displayed: **“Delete current head shape points?”**

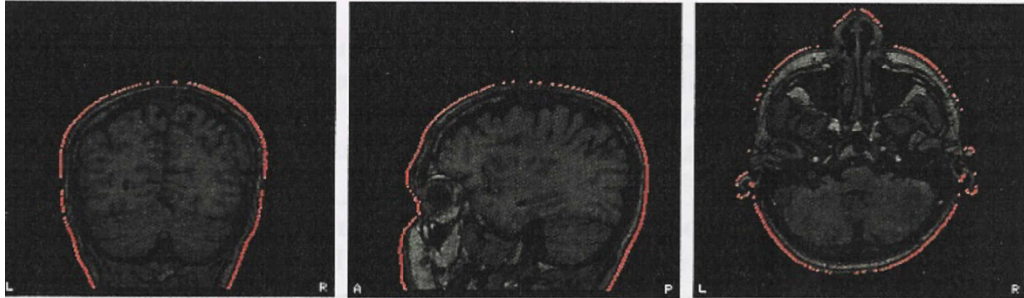
- Click **Yes** – to delete the current list of head shape points and display the Open Head Shape file dialog.
- Click **No** – to keep the current list of head shape points and to display the Open Head Shape file dialog.

—If more head shape data is loaded, the data is added to the current list of points.

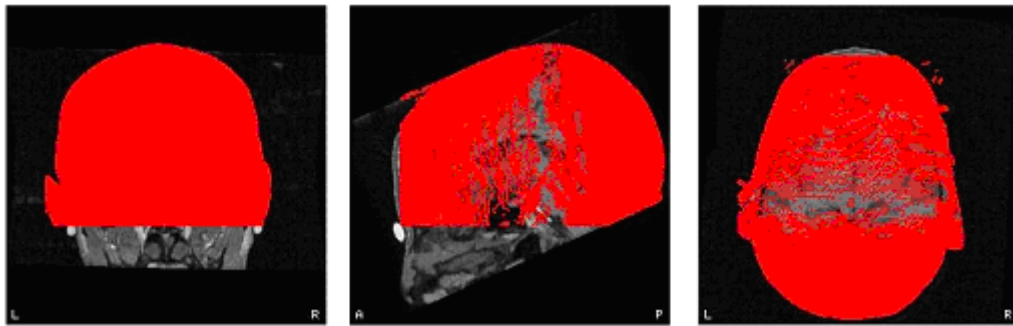
—If the fiducial points do not match, you are given the choice to:

- ❖ **Update the fiducial points** – replace the current fiduciary points with those from the head shape data file or
- ❖ **Re-align the fiducial points** – keep the current fiduciary points and realign the new head data points to the current head coordinate system.

5. If the *.shape_info file is missing the following dialog is displayed:
“No fiduciary points for head shape. Cannot verify alignment.”
- Click **Ok** – to load the head shape data and use the current head coordinate system.
 - Click **Cancel** – if you do not want to load the data.



Head Shape Points for Slices Displayed.



All Head Shape Points Projected Onto Displayed Slices With Excluded Points.

6. Click on the **File>Save Head Shape** menu to save.
- Click on the **Save points** in menu to select the coordinate system:
 - Head coordinates or
 - MRI coordinates
 - Click **OK**.
7. If points have been excluded by selecting any of the **Exclude Points** check boxes, the following choice is displayed: **“Save excluded points?”**
- Click **Yes** – to save all available head shape points, including the points excluded in the **Exclude Points** check boxes.
 - Click **No** – to save only points that have not been excluded.
8. **Close > Exit**.

Source Modeling /

SAM Analysis

SAM is an acronym for “synthetic aperture magnetometry”. It is an analysis method for imaging electrophysiological brain activity. A SAM volume, which is a color-coded volumetric image, is created using samCov. SamCov computes the covariance. For single state tests, samCov computes the mean covariance of all specified MEG trials and one covariance file will be generated. For two-state tests, samCov computes the mean covariance of the active and of the control trials, as well as the mean covariance of the pooled active and control data. This results in three covariance files being generated – one each for the active, control, and their pooled mean. SAM covariance files are given a suffix “cov.” appended to their name.

The input parameters to samCov specify the bandwidth and (optionally) time segments to be extracted from the MEG data, for covariance calculation. SAM volumes cannot be created in MRIVIEWER but are instead computed using SAM software and loaded into MRIVIEWER or AFNI. SAM volumes cannot be edited or saved using MRIVIEWER. However, peaks may be marked and the image’s bounding coordinates, brightness, transparency and level of detail are adjustable.

The SAMsuite includes eight command line programs for computing parametric images, using exact solutions for T, F, and Z-statistics. SamCov must be performed first. SamCov is intended for use with samSrc, linSrc, or musicSrc.

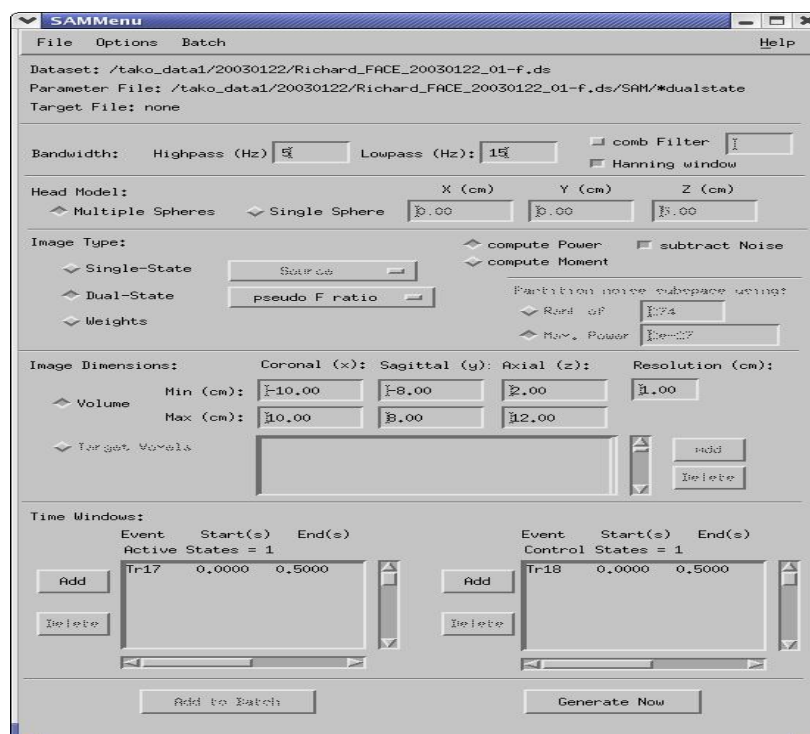
I. CREATING A SAM VOLUME (SAMCOV):



Note: samCov does not inherit any processing parameters from DataEditor.

A. Specifying Parameters

1. From a terminal window, type **SAMMenu** <return>.
2. Click on **File > Open Dataset**.
3. Select the **Directory** and **Dataset**. Use filtered data for this.
4. You may be asked a few questions:
 - a. If asked, “SAM subdirectory does not exist. Create it?” click **Yes**.
 - b. If asked, “Multisphere file does not exist for this dataset. Do you want to create one from a head shape file?” choose accordingly (user determined). If you choose **Yes**, select the appropriate directory and head shape file.
5. At the SAM Menu select the parameters:
 - Bandwidth,
 - Head Model,
 - Image Type,
 - Image Dimension



SAM Menu

6. Enter the **Time Window** parameters (If **Dual –State Image Type** was selected then **Active State** and **Control State** parameters must be entered):

Active States window (left side):

- a. Click **Add** to bring up the popup **Add Time Window**.
- b. Pick the appropriate marker by clicking on **(TIME-ZERO)**.
- c. Enter values for **Window Start** and **Window Stop**.
 - The selected marker is always at time zero so to select a time interval that begins before the marker (pre- event),
 - The Start time (in seconds) must be negative.
 - The Stop is how much time after each marker the calculations will run (post -event).
- d. Click **OK**.

Control States window (right side):

- a. Click **Add** to bring up the popup **Add Time Window**.
- b. Pick the appropriate marker by clicking on **(TIME-ZERO)**.
- c. Enter values for **Window Start** and **Window Stop**.
 - The selected marker is always at time zero so to select a time interval that begins before the marker (pre- event),
 - The Start time (in seconds) must be negative.
 - The Stop is how much time after each marker the calculations will run (post-event).
- d. Click **OK**.



Note: The **Window Start** and **Window Stop** must be the same for both the **Active States** and the **Control States**.

7. After entering all parameters, click **File>Save Parameter File As** to save the settings.
8. Click **Generate Now**.
9. If the parameter file has not been saved, a pop up dialog will prompt you to so.
10. At the prompt “**Execute SAM commands with current settings?**” click **Execute Now**.

SAM Parameters

Bandwidth	High Pass (Hz):	The frequency range of interest. User determined.
	Low Pass (Hz):	The frequency range of interest. User determined.
	Comb Filter:	<ul style="list-style-type: none"> Designates a comb-notch filter fundamental frequency (Hz). This flag and its parameter is optional. The default operation is no comb-notch filtering. When selected, the frequency lines for the specified frequency and all harmonics up to half the sample rate are removed from the spectrum, prior to computing the signal covariance. Do not use this flag if you are generating SAM weighting coefficients in a bandpass including the power mains frequency. Doing so will disable SAM's ability to attenuate this interference in displays of virtual sensor time-series or source movies. The comb-notch should only be used when generating static SAM images.
	Hanning Window:	<ul style="list-style-type: none"> Tapers the data to zero at the start and end of each selected time segment. This reduces leakage of signal into adjacent frequencies, providing the most accurate bandpass control. However, it also tends to emphasize MEG signals at the center of each data window, and de-emphasize signals at the ends. Use of the <u>hanning window</u> is therefore highly recommended. To compensate, the user should extend the time duration of each data window, to reduce signal loss.
Head Model	Multiple Sphere:	<ul style="list-style-type: none"> Creates multiple local spheres to generate a realistic head-shape model; Reads the head boundary file (head shape file) generated by MRIVIEWER, together with the sensor positions (in the head frame) from a select dataset. It then determines a best-fit local sphere for every SQUID sensor – primary and reference sensors. Computes the active state versus noise. The multiple local spheres provides an improved forward solution over the single local sphere method. Use the default settings.
	Single Sphere:	<ul style="list-style-type: none"> Creates a single sphere.
Image Type	Single-State:	<ol style="list-style-type: none"> a. Source – Calculates power at each voxel without removing noise interference. b. Pseudo Z Deviate – Calculates power at each voxel while removing noise interference. c. MUSIC Metric – This is not a SAM, and involves the multiple dipole fit method. It is a Multiple Signal Classification, an algorithm which computes static images from the covariance statistics. Both single and two-state images modes are supported.

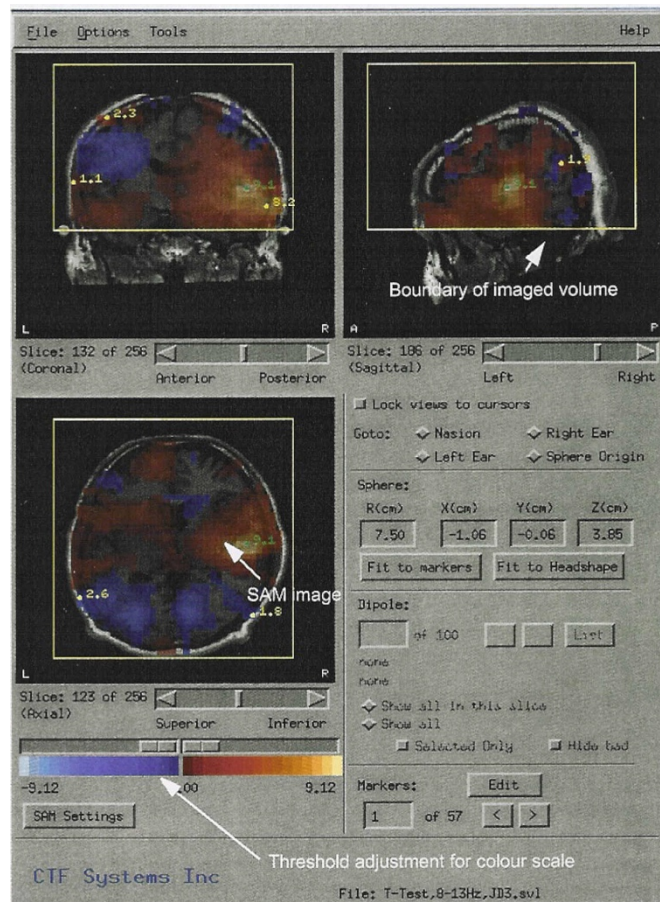
	Dual-State:	<ul style="list-style-type: none"> a. Source – Subtracts the control power from the active, without removing noise interference. b. Pseudo Z Deviate – Subtracts the control from the active while removing noise interference. c. Pseudo Student's T – Subtracts control from the active and divides the difference by a standard error pooled across the active and control. d. Pseudo F Ratio – Divides the active power by the control power. e. MUSIC Metric – This is not a SAM, and involves the multiple dipole fit method. It is a Multiple Signal Classification, an algorithm which computes static images from the covariance statistics. Both single and two-state images modes are supported.
	Weights:	Outputs optimal spatial filter for a virtual channel.
	Compute Power	Defines the variance (in Tesla) for dividing the covariance matrix into noise and signal subspaces. This is the typical choice.
	Compute Moment	Computes the moment instead of the power. Note that the typical choice is to compute the power.
	Subtract Noise	Removes noise interference.
Image Dimension	Volume:	<ul style="list-style-type: none"> • Min (cm) Coronal(x), Sagittal (y), Axial (z) • Max (cm) Coronal(x), Sagittal (y), Axial (z)
	Resolution (cm):	<ul style="list-style-type: none"> • The fineness of detail that be distinguished in the shaded area of activity representing synchronization (red) or desynchronization (blue); voxel size. • Recommended range: < 1.0 cm, but a typical resolution is 0.75. •
	Target Voxels:	When Weights is selected: <ul style="list-style-type: none"> • Click Add and • Enter the coordinates to target virtual channels. • Click OK.
Time Window	Event Active State=	Event Active States = 1 Start (s) – the time interval before the marker (pre event window); designated by negative numbers; the event represents zero (0). End(s) – the time interval after the marker (post event window); designated by positive numbers; the event represents zero (0).
	Event Control State=	Event Control States = 1 Starts(s) – the time interval before the marker (pre event window); designated by negative numbers; the event represents zero (0). End(s) – the time interval after the marker (post event window); designated by positive numbers; the event represents zero(0).
Add to Batch		Refer to user manual.
Generate Now		Creates the SAM Volume file (.svl).

B. Opening a SAM Volume

1. Type “**MRViewer**” at a terminal window to open the program or in MRViewer, click **File > Open MRI File**.
2. Select the appropriate MRI file. Click **OK**.
3. Click **File > Open SAM File...** to open a SAM Volume.
4. Open the appropriate dataset (the same file opened in SAMMenu).
5. Within the dataset, open the SAM directory.
6. Under the Files window, click on the new MRI SAM Volume file (.svl file).
7. Click **OK**.

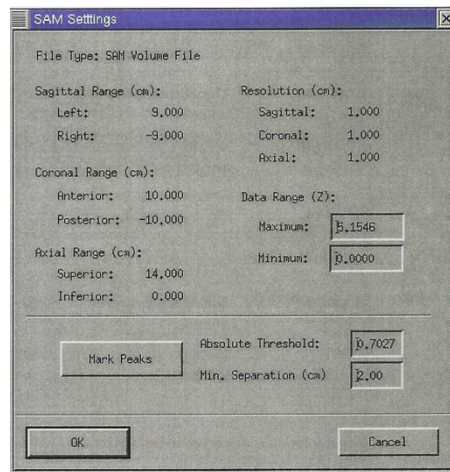
C. Thresholding the SAM Volume

The SAM volumetric images are superimposed on individual MRI slices. The SAM volume outline is only displayed if the MRI has been orthogonalized. Otherwise, no outline is shown. The Sam intensities are contained in a volume that is bounded. The boundary is displayed as a yellow line in all three orthogonal images.



MRViewer Displaying Color SAM Image.

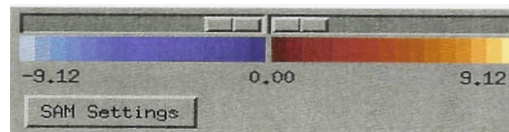
1. After opening the image, the current maximum intensity level should be adjusted in the **Maximum** field from the **SAM Settings Dialog**.



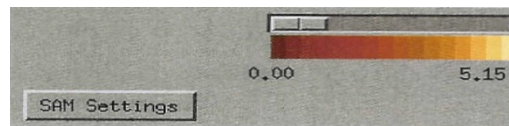
[SAM Settings Dialog](#)

2. The **Slider Bar** located above each color gradient allows you to control the number of colors displayed on the SAM image overlay. The location of the slider bar along the color gradient defines the threshold. When a SAM image is loaded, the Sam controls appears in the main window.

- By moving the slider bar away from 0, the image voxels representing values smaller than the threshold are not displayed.
- In this way, only key features represented by large absolute values are visible on the Sam image overlay, allowing the MRI to be more visible.
- The image is displayed in color – blue for negative values and red for positive values.
- When the SAM image contains both positive and negative values both red (positive) and blue (negative) controls will be available.
- When the SAM image contains only positive values only that portion of the slider bar controls are available.



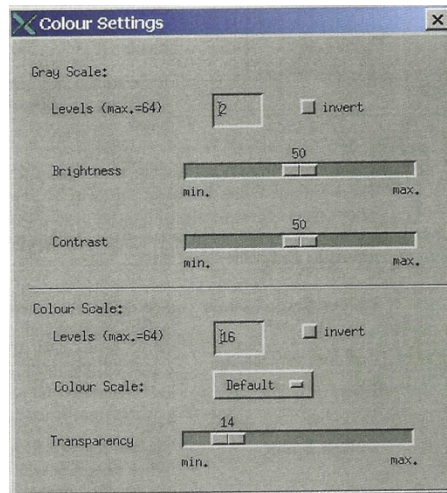
SAM controls for an image with both positive and negative values (Dual states).



SAM controls for an image with only positive values (Single states).

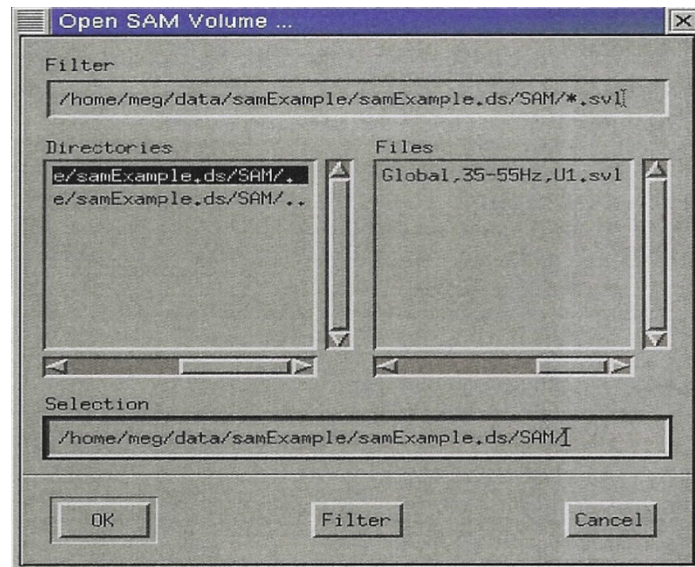
3. Adjust the Color Settings. The Brightness, Contrast, Color Scale and Transparency can be changed by accessing the **Options > Color Options** menu.
 - Use the Transparency setting (**Options > Color Options > Color Settings dialog**) to adjust the SAM image to view both MRI and SAM volume effectively.

- The **Transparency Slider** can be used to make the MR Image visible through the SAM overlay image. For SAM images with higher or lower resolution than the original MR image, the SAM slice coinciding or closest to the currently viewed slice will be displayed.

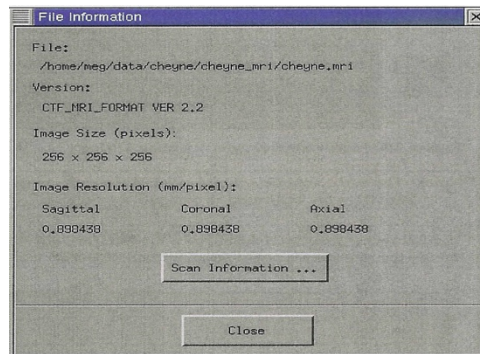


[Color Settings Dialog](#)

1. From MRIVIEWER click on **File > Open Sam File.**
(or from the Tools menu select MRIVIEWER).



2. Double click on the appropriate file under Directories.
3. Then double click on the appropriate dataset under Files.
4. **SAM Settings > File Information** dialog displays. This dialog displays the file and scan information of the currently specified MRI.
5. Click on **Close.**



File Information Dialog

6. MRIVIEWER loads the selected file with the SAM image.

II. SAM Statistical Analysis Command Line Programs Options:

There are eight command line programs including samCov for computing statistical analysis. All of these programs are invoked by their name, followed by all command-line flags and their respective arguments on a single line.



SamCov must be performed first.

1. Type “**samCov <dataset>**” at the terminal window.

SAM Suite		
samCov	*Mandatory for basic SAM analysis	<ul style="list-style-type: none">• Computes the <u>covariance</u> of a single dataset. Operates on either un-averaged or averaged MEG data.• The input parameters specify the bandwidth and (optionally) time segments to be extracted from the MEG data, for covariance calculation.• If all time samples are to be integrated, then no additional files are needed. If specific time windows or active and control states are designated, then the file containing these specifications must be given as an argument to samCov.
samSrc	*Mandatory for basic SAM analysis	<ul style="list-style-type: none">• Computes either <u>static source image or coefficient files</u> (for virtual channels and source movies).• Uses the covariance files created by samCov.
linSrc		<ul style="list-style-type: none">• Computes static source images using the <u>linear minimum variance beamformer method</u>. It is analogous to samSrc.• Requires the covariance files from the samCov.

SAM Suite (continue)

samJCov	Yields exact statistical calculations of source probabilities	<ul style="list-style-type: none"> • Computes <u>multiple covariance matrices</u> – one per data segment – using the <u>jackknife statistical method</u> (a statistical resampling method). • Used in place of samCov, when exact <u>Z, T or F-statistics for source images</u> are required. The resulting statistics account for the variability in brain activity across multiple trials, within a single dataset. • Operation of samJCov on signal averaged data is meaningless, since multiple trials are required.
samJSrc	Yields exact statistical calculations of source probabilities	<ul style="list-style-type: none"> • Computes <u>static source images</u> using the <u>multiple covariance files</u> generated by samJCov.
linJSrc		<ul style="list-style-type: none"> • Computes <u>static source images using the multiple covariance files</u> generated by samJCov.
musicSrc	Uses part of the SAM interface	<ul style="list-style-type: none"> • Computes static images of the <u>MUSIC (multiple signal classification) metric algorithms</u>. • Uses the covariance statistics generated by samCov. • Both single and two-state image modes are supported.
localSpheres	Optional - Without them, samSrc is able to accept a single best-fit sphere origin as a command line parameter. Use of the multiple local sphere information is, however, highly recommended.	<ul style="list-style-type: none"> • The purpose of localSpheres is to generate a realistic head-shape model, based upon multiple local spheres. • LocalSpheres reads the head boundary file (head shape file) generated by MRIVIEWER, together with the sensor positions (in the head frame) from a selected dataset. • Then determines a best-fit local sphere for every SQUID sensor – primary and reference sensors. Any head boundary file may be used by digitizeHead. However that file must be formatted in MRIVIEWER. • Multiple local spheres provide an improved forward solution over the single local sphere method.

*samCov is intended for use either with samSrc, linSrc, or musicSrc.

**samJCov is intended for use with samJSrc or linJSrc.

Appendix

File Paths and Name Conventions

Extension	Type	Description
.ds	Dataset	Contains data that can be viewed in DataEditor.
.mrk	Marker File	Contains marker information created in DataEditor.
.mri	MR Image File	A converted MRI file that should be opened in MRIViewer.
.dip	Dipole File	Contains dipole information created in MRIViewer.
.shape	Head Shape File	Contains head shape information created in the Head Shape window in MRIViewer.
.shape_info	Shape Information File	Contains fiduciary points and coordinate frame.
.hdm	Head Model File	Contains “best-fit” sphere information.
.de	Template Setup File	Contains template setup information for DataEditor.
.dm	Template File	Contains template information for DataEditor.
.txt	Text File	Contains information in text format.
.svl	SAM Volume File	Contains SAM parameters and calculations created in SAMMenu
.fid	Fiduciary File	Contains information on Fiduciary Markers.
.cov	SAM Covariance File	Covariance information created in samCov.
.vc	Virtual Channels File	Contains virtual channel definitions.
.cfg	Configuration File	Contains channel information saved in the Channel Selector in DataEditor.
.wts	SAM Weight File	SAM coefficient information.
.meg_config	MEG Configuration File	SAM user-defined channel sets.
.ps	Plot Tool File	Plot information created in the Plot Tool.
.cls	Class File	Class Information.
.hdr .img	Analyze	Contain MRI data.

File Paths and Name Conventions (continued)

.rp	Parameter File	Reads parameters.
.acq.rp	Default Protocol File	
(a text file)	Polhemus	List of head shape points relative to the MEG Head Coordinate System.
(a text file)	Head Shape Import	List of head shape points relative to the MEG Head Coordinate System.
-f	Filtered Data	'-f' is placed at the end of a data filename to indicate that the data has been filtered.
-av	Averaged Data	'-av' is placed at the end of a data filename to indicate that the data has been averaged.
D2	Dual-State	Denotes that two-state imaging using the pseudo T statistic option was selected in SAMMenu.
D3		Denotes that three-state imaging using the pseudo F statistic option was selected in SAMMenu.

INDEX

A

ADC Markers	3
Accessing Data	3, 4, 7, 44, 48
Accessing Raw Data	7
Active States	51, 53
Adcs Channels	11
Addmarker Command Line Program Options	3, 6, 18
AFNI	3, 6, 7, 43, 50
Aligning / Matching Head Shape With Fiducials	7
Amplitude, Threshold	29
Analyse Menu	13
Average Trials	5
Averageds Command Line Program	33
Averager Program	3, 6, 27
Averaging Data	3, 6, 27
Averaging While Excluding Eye Blinks	3, 6, 29

B

Band Pass Filter	9
Basic Signal Processing	9
Best-Fit Sphere Model	4, 38

C

Channel Selector	8, 11, 19, 29, 31
Class Dialog	23, 25
Classify Trial As	13, 19
Classify Trials	19
Classifying / Excluding A Single Channel Using DataEditor3	18
Classifying / Excluding An Entire Trial Using DataEditor	3, 19
Classifying / Excluding Eye Blinks Using MATLAB/ICA	16
Classifying Trials (Using Single Channel Display)	25
Command Line Programs	11, 50, 58
Control States	51, 53
Control Toolbar	8
Co-Register The MEG	5
Co-Registering Data / Set Fiducials	6
Co-Registering Data Using Fiducials	4
Create A Head Shape File	5
Create A Realistic (Best-Fit) Head Model	5
Create Head Model	7
Create Head Shape	7
Creating / Identifying ADC Markers Using Threshold Detector	11
Creating / Identifying Stim Markers	3
Creating A Head Shape	4
Creating A SAM Volume	3, 41
Creating Markers / Identifying ADC Markers Using Threshold Detector	6
Creating Markers / Identifying Stimulus Markers Using Dataeditor	6
Creating/Identifying Stimulus Markers (Trigger Markers) Using Dataeditor	16
Creating/Modifying A Classification	23

D

Data Conversion	3
Dataeditor ... 3, 5, 6, 7, 11, 13, 16, 22, 23, 25, 27, 28, 29, 30, 31, 50	
Dataset Selector	7, 8
Dead Time	13, 19, 29
Derivative Threshold	29

E

ECD	34
Edge Enhancement	4, 44
Edit Markers Button	16
Editing Data	3
Editing Data	6, 11
Editing Parameters	3, 7
EEG/ADC Amplitude Threshold	13, 19
EEG/ADC Derivative Threshold	13, 19
EEGLAB	3, 20, 21
Environmental	7, 9
Equivalent Current Dipole	34
Erosion	4, 44
Exclude Bad Trials	28
Excluding / Removing Eye Blink Artifact	6
Excluding / Removing Eye Blink Artifacts Using MATLAB & EEGLAB	20
Excluding A Single Channel	6
Excluding An Entire Trial	6
Eye Blinks	5, 7, 19, 20, 29, 30, 33

F

Filter Data	5
Filtering Data	3
Filtering Data	6, 9
Flow Chart	3, 6

G

Gain Control	8
Generate Average	28
Generate Data Covariance Files	6
Generate Markers	5
Generate SAM Volumes	6
Generate Standard Deviation	28

H

Head Coordinate System	34, 39, 48, 49
High Pass Filter	9

I

ICA	3, 6, 20, 21
Identifying Markers For The Adcs	11
Import And Create A MEG	5
Import And Display Color-Coded "Volumetric Images"	6
Importing	4, 34
Importing Mri Files	34
Importing The	4
Independent Component Analysis	11
Inspect, Edit, And Process	5
Integrating MEG&MRI	4, 31

L

Low Pass Filter	9
-----------------------	---

M

Magnetic Resonance Imaging	34
Manual / Trigger Option	16
Manual Markers	11
Manual Markers	16

Marker Dialog.....	16
Marker Sets.....	16
Marking / Removing Eye Blinks Or Other Artifact Using Threshold Detector	6
Marking Eye Blinks Or Other Artifacts	19
MATLAB.....	3, 6, 20, 21
MIPAV	37
MRI.. 3, 4, 5, 6, 34, 35, 36, 37, 38, 39, 40, 43, 44, 45, 47, 49, 54, 55, 57	
MRI Data Conversion	6, 34
Mriconverter	5, 34

N

Noise Reduction	9
Notch Filter.....	9

O

Offset And Trend Removal.....	9
Opening The SAM Volume.....	3, 7
Orthogonalize MRI.....	6, 34
Other Analysis Methods	<i>See Flowsheet</i>
Overlay.....	28, 55, 56

P

Physiological Artifacts.....	7
Post Event.....	28
Powerline Filter	9
Pre Event	28
Previewing & Specifying Parameters.....	4
Previewing & Specifying Parameters.....	6
Previewing Mri Data & Specifying Input Parameters	37
Processing Parameters	9

R

Review, Edit & Process	3
Review, Edit & Process MEG Data	7

S

SAM Analysis.....	3, 50
SAM Statistical Analysis Command Line.....	58
SAM Statistical Analysis Command Line Programs.....	7
Samcov	3
Save Marker As	13, 19
Segmentation	4, 7, 44
Showing All Points	4
Single Channel Display	25, 26
Source Modeling	3
Source Modeling / Sam Analysis	7
STIM.....	11
Stimulus Markers	11, 18
Subject Prep	60
Surface Extraction	4, 44

T

Target Event	28
Terminal Window 7, 18, 27, 29, 31, 33, 34, 39, 43, 44, 50, 54, 58	
Threshold Detect Parameters	13
Threshold Detect Program	11, 19
Threshold Detect Program	13
Thresholding	3, 4, 7
Trial Classification.....	23
Trigger Markers.....	11
Trigger Markers.....	16
Types Of Markers	7, 11